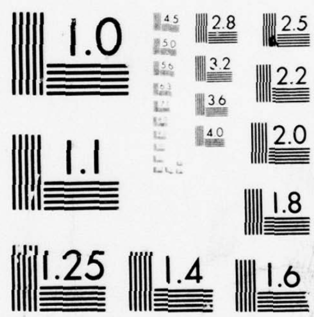
AD-A068 250 CALIFORNIA UNIV BERKELEY DEPT OF MECHANICAL ENGINEERING F/G 13/8
ANALYSIS OF AXISYMMETRIC SHEET-METAL FORMING PROCESSES BY THE R--ETC(U)
SEP 78 F33615-77-C-5111

UNCLASSIFIED AFML-TR-78-120 NL

1 OF 2
AD
A068250


1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0

AFML-TR-78-120

LEVEL II

2

ANALYSIS OF AXISYMMETRIC SHEET-METAL FORMING BY THE RIGID-PLASTIC, FINITE-ELEMENT METHOD

MECHANICAL ENGINEERING
UNIVERSITY OF CALIFORNIA
BERKELEY, CALIFORNIA

SEPTEMBER 1978

TECHNICAL REPORT AFML-TR-78-120
Technical Report May 1977 – May 1978

Approved for public release; distribution unlimited.

AIR FORCE MATERIALS LABORATORY
AIR FORCE WRIGHT AERONAUTICAL LABORATORIES
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433

NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

This report has been reviewed by the Information Office (OI) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

This technical report has been reviewed and is approved for publication.

Harold L. Gegel

HAROLD L. GEGEL
Project Engineer

FOR THE COMMANDER

Norman M. Geyer

NORMAN M. GEYER
Acting Chief
Processing and High Temperature Materials Branch
Metals and Ceramics Division

Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.

AIR FORCE/56780/6 April 1979 — 315

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE	READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER: 18 AFML-TR-78-120 / 19 TR-78-120 — 2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle): 6 ANALYSIS OF AXISYMMETRIC SHEET-METAL FORMING PROCESSES BY THE RIGID-PLASTIC, FINITE-ELEMENT METHOD.	5. TYPE OF REPORT & PERIOD COVERED: 9 Technical Report. June 1977 - June 1978.
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s): 10 J. H. Kim, S. I. Oh and Shiro Kobayashi	8. CONTRACT OR GRANT NUMBER(s): 15 F33615-77-C-5111 new
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Department of Mechanical Engineering, University of California, Berkeley, California 94720	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: 62102F 16 2418 17 04 241804 24180406
11. CONTROLLING OFFICE NAME AND ADDRESS: Air Force Materials Laboratory, Air Force Systems Command, Wright-Patterson Air Force Base, Ohio 45433	12. REPORT DATE: 11 September 1978
	13. NUMBER OF PAGES: 141
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office): 12 148 p.	15. SECURITY CLASS. (of this report): Unclassified
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release: distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Sheet metal forming, mechanics, finite element method, punch stretching, hydraulic bulge, cup drawing. Rigid-plastic analysis.

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report describes the development of a finite-element model for analyzing sheet-metal forming processes. Materials are assumed to be rigid-plastis with the view that the usefulness of an analysis method depends largely upon solution accuracy and computation efficiency. First, the variational formulation applicable to sheet-metal forming is described by considering solution uniqueness and the

(over)

over

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-LF-014-6601

400 426 slt

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

79 05 03 110

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

effect of geometry change involved in the forming processes. From this variational formulation, a finite-element process model based on the membrane theory is developed. Then, three basic sheet-metal forming processes, namely, the bulging of a sheet subject to hydrostatic pressure, the stretching of a sheet with a hemispherical head punch, and deep drawing of a sheet with a hemispherical head punch, are solved. The solutions arrived at by the rigid-plastic, finite-element method are compared with existing numerical solutions and the experimental data. The agreement is generally excellent and it is concluded that the rigid-plastic, finite-element method is efficient for analyzing sheet-metal forming problems with reasonable accuracy.

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

TABLE OF CONTENTS

SECTION		PAGE
I	INTRODUCTION	1
II	BACKGROUND	3
	1. Uniqueness	3
	2. Geometry change	7
III	FORMULATION	13
	1. Variational formulation	13
	2. Theory of the finite-element method	14
	3. Modeling of axisymmetric problems	17
	4. Linearization	22
IV	HYDROSTATIC BULGING	23
	1. Introduction	23
	2. Computational procedures	28
	3. Results and discussion	29
V	STRETCHING OF A SHEET WITH HEMISPHERICAL PUNCH	37
	1. Introduction	37
	2. Computational procedures	40
	3. Results and discussion	46
VI	DEEP DRAWING OF A SHEET WITH HEMISPHERICAL PUNCH	68
	1. Introduction	68
	2. Computational procedures	69
	3. Results and discussion	72
VII	SUMMARY AND DISCUSSION	79
	Appendix A: PROGRAM FOR THE INITIAL GUESS FOR HYDROSTATIC BULGING ANALYSIS	83
	Appendix B: PROGRAM FOR THE ANALYSIS OF HYDROSTATIC BULGING	86
	Appendix C: PROGRAM FOR THE ANALYSIS OF PUNCH STRETCHING	100
	Appendix D: PROGRAM FOR THE ANALYSIS OF DEEP DRAWING AND PUNCH STRETCHING WITH ROUND DIE CORNER	118
	REFERENCES	138

LIST OF ILLUSTRATIONS

FIGURE		PAGE
1	Approximation of the sheet geometry into a series of conical frustra	20
2	Schematic view of hydrostatic bulging	27
3	Hydrostatic pressure vs. polar thickness strain	30
4	(a) and (b) Distribution of strains	31
5	Distribution of stresses	32
6	Bulge profile	32
7	Polar height vs. pressure	34
8	Bulge profile	35
9	(a) Circumferential strain distribution	36
	(b) Thickness strain distribution	36
10	Schematic view or the stretching of a sheet with a hemispherical head punch	38
11	Geometrical requirement for the node on the contact region	41
12	Punch head vs. punch travel	48
13	Thickness strain distribution	49
14	Circumferential strain distribution	50
15	Thickness strain distribution	50
16	Effect of step size	52
17	Effect of mesh size	52
18	Distribution of thickness strain when die profile is considered	54
19	Distribution of circumferential strain when die profile is considered	54

FIGURE PAGE

20 Comparison of the numerical solution with the experimental data for circumferential strain distribution. 56

21 Comparison with the experiment for thickness strain distribution. 56

22 Stress-strain curve for al. 2036-T4 58

23 Experimental (Johnson's wax as lubricant) and theoretical (μ = 0.2) strain distributions for punch size (r_p/r_0 = 0.75/0.80). (a) Thickness strains; (b) Circumferential strains 61

24 Experimental (Johnson's wax as lubricant) and theoretical (μ = 0.2) strain distributions for punch size (r_p/r_0 = 0.45/0.80). (a) Thickness strains; (b) Circumferential strains 62

25 Comparison of theoretical thickness strain distributions using (1) the parabolic workhardening law and (2) the Voce equation for punch size (r_p/r_0 = 0.75/0.80) with (a) μ = 0 and (b) μ = 0.2 64

26 Comparison of theoretical thickness strain distributions using (1) the parabolic workhardening law and (2) the Voce equation for punch size (r_p/r_0 = 0.45/0.80) with (a) μ = 0 and (b) μ = 0.2 65

27 Comparison of theoretical load displacement curves using (1) the parabolic workhardening law and (2) the Voce equation for μ = 0. 66

28 Comparison of theoretical load displacement curves using (1) the parabolic workhardening law and (2) the Voce equation for μ = 0.2. 67

29 Schematic view of deep drawing of a sheet with a hemispherical head punch. 70

30 Distribution of thickness strain for μ_p = 0.04, μ_d = 0.04 . . 74

31 Distribution of circumferential strain for μ_p = 0.04, μ_d = 0.04 75

32 Distribution of thickness strain for μ_p = 0.1, μ_d = 0.04. . . 76

33 Distribution of circumferential strain for μ_p = 0.1, μ_d = 0.04 77

34 Punch load vs. punch depth. 77

SECTION I

INTRODUCTION

The metal forming processes basically involve large amounts of elastic deformation, and, due to the complexities of plasticity, the exact analysis of a process is infeasible in most of the cases. Thus, a number of approximate methods have been suggested, with varying degrees of approximation and idealization. Among these, techniques using the finite-element method take precedence because of their flexibility, ability to obtain a detailed solution, and the inherent proximity of their solutions to the exact one.

A prime objective of mathematical analysis of metalworking processes is to provide necessary information for proper design and control of these processes. Therefore, the method of analysis must be capable of determining the effects of various parameters on metal flow characteristics. Furthermore, the computation efficiency, as well as solution accuracy, is an important consideration for the method to be useful in analyzing metalworking problems.

With this viewpoint in mind, successful efforts have been carried out in analyzing various deformation processes, such as compression, heading, piercing, extrusion and drawing by the rigid-plastic, finite-element method (matrix method) [1]-[7].

The formulation of the matrix method, however, cannot be extended to the sheet-metal forming analysis due to the following reasons:

(1) The classical variational formulation which is the basis of the matrix method does not necessarily determine a unique deformation mode. Physically, there is no inherent indeterminacy for work-hardening solids, but this indeterminacy is due rather to the fact

that the workhardening rate is not included in the mathematical formulation of the classical variational principle.

(2) The kinematic assumption in the matrix method is not longer valid for the sheet-metal forming process. As long as bulk deformation or in-plane stretching are concerned, this kinematic assumption that the magnitude of the rate of rotation is negligible compared to the strain rate does not deviate much from the real situation and yields solutions consistent with reality. Geometric nonlinearity in sheet-metal forming, however, invalidates such a simplification.

The objective of the present investigation is, therefore, to develop and establish a finite-element method for sheet-metal forming processes.

In Section II various forms of variational formulations are reviewed in the light of uniqueness and geometry change which leads to a realization of the necessity of new formulations. In Section III a new formulation is obtained and the development of the finite-element model from it is described. With the particular example of sheet-metal forming processes in mind, the idealization of plane stress state and membrane theory is implemented. Furthermore, the development is confined to the case of axisymmetrical problems.

To establish the validity of the proposed method, three basic sheet-metal forming processes are analyzed and the solutions are compared with other available experimental data and numerical solutions. Hydrostatic bulging is treated in Section IV. Punch stretching with a hemispherical punch is discussed in Section V. To make the problem tractable, one moving contact boundary is considered first by neglecting die profile; then the analysis is extended to include two moving boundaries. In Section VI deep drawing with a hemispherical punch is solved.

SECTION II

BACKGROUND

1. Uniqueness

We consider the quasistatic deformation of a rigid-plastic solid. On a portion S_v of the surface S of this body are prescribed given velocities, while the remainder S_T of the surface S is subjected to given surface tractions T_i. Assuming that these surface velocities and tractions are such that the entire body is in a state of plastic flow, we want to determine the stresses σ_{ij} and strain rates $\dot{\varepsilon}_{ij}$ throughout the body.

The conventional formulation of variational principle for this problem is that among all kinematically admissible strain rate fields $\dot{\varepsilon}^*_{ij}$, the actual one minimizes the expression (Hill [8]),

$$\pi_1 = \int \bar{\sigma}\dot{\bar{\varepsilon}}^* \, dv - \int_{S_T} T_i v_i^* \, dS, \tag{1}$$

where $\bar{\sigma}$ is the effective stress, $\dot{\bar{\varepsilon}}$ is the effective strain rate defined by

$$\bar{\sigma} = \sqrt{\frac{3}{2}}\sqrt{\sigma'_{ij}\sigma'_{ij}},$$

$$\dot{\bar{\varepsilon}} = \sqrt{\frac{2}{3}}\sqrt{\dot{\varepsilon}_{ij}\dot{\varepsilon}_{ij}},$$

respectively, where σ'_{ij} is the deviatoric component of σ_{ij}. Here a strain rate field $\dot{\varepsilon}^*_{ij}$, defined throughout the body under consideration, is called kinematically admissible if it is derivable from a velocity field v_i^* which satisfies the condition of incompressibility $v^*_{i,i} = 0$[†] throughout the body

---

[†]The comma denotes the differentiation with respect to coordinates, e.g.,

$$v_{i,i} = \frac{\partial v_i}{\partial x_i}.$$

and the boundary conditions on S_v. The variational principle in this form has been successfully applied to the analysis of metal forming problems, such as extrusion [6]. As was found out later, and we will discuss this shortly, the success is related to the type of boundary conditions prescribed on the surface of the body undergoing deformation. In general, with the variational formulation of π_1 in Eq. (1), there is a question regarding uniqueness of deformation mode even though the stress field is uniquely determined [8], [9].

Consider an incipient flow in a rigid-plastic solid, workhardening or perfect plastic, governed by the following partial differential equations which are, of course, dual to the variational formulation π_1. With respect to Cartesian reference frame x_i the following equations hold:

<u>Equilibrium equations</u>

$\sigma_{ij,j} = 0$ in the absence of body force (2a)

<u>Strain rate-velocity relationship</u>

$\dot{\varepsilon}_{ij} = \frac{1}{2}(v_{i,j} + v_{j,i})$ (2b)

<u>Constitutive equation</u>

$\mu\sigma'_{ij} = \dot{\varepsilon}_{ij}$, μ being an arbitrary constant (2c)

<u>Yield criterion</u>

$\bar{\sigma} = \sqrt{\frac{3}{2}}\sqrt{\sigma'_{ij}\sigma'_{ij}} = H(\bar{\varepsilon})$, where $\bar{\varepsilon}$ is the effective strain defined by $\bar{\varepsilon} = \int d\bar{\varepsilon}$ if $d\bar{\varepsilon} = \sqrt{\frac{2}{3}}\sqrt{d\varepsilon_{ij}d\varepsilon_{ij}}$ (2d)

<u>Boundary conditions</u>

$n_j\sigma_{ij} = \hat{T}_i$ on S_T, (2e)

$v_i = \hat{v}_i$ on S_v, where n_j is the unit normal vector to the surface of the body; $\hat{T}_i$ and $\hat{v}_i$ are prescribed values

Suppose that $(\sigma_{ij}^{(1)}, \dot{\varepsilon}_{ij}^{(1)})$ is the solution to this boundary value problem. Construct a different set of stress fields and strain rate fields $(\sigma_{ij}^{(2)}, \dot{\varepsilon}_{ij}^{(2)})$, where $\sigma_{ij}^{(2)} = \sigma_{ij}^{(1)}$, $\dot{\varepsilon}_{ij}^{(2)} = C\dot{\varepsilon}_{ij}^{(1)}$. C is any arbitrary factor and may vary from point to point throughout the body. Then, it is easily shown that this set $(\sigma_{ij}^{(2)}, \dot{\varepsilon}_{ij}^{(2)})$ satisfies all the governing equations except for the boundary conditions on S_v. On S_v the velocity integrated from $\dot{\varepsilon}_{ij}^{(2)}$ should coincide with the prescribed value $\hat{v}_i$. Since strain rate-velocity relation is linear, integrating $\dot{\varepsilon}_{ij}^{(2)}$ would yield $C\hat{v}_i$ if $\dot{\varepsilon}_{ij}^{(1)}$ is integrated to give $\hat{v}_i$, and therefore C must be unity on S_v. With this and the compatibility requirement the deformation mode may or may not be uniquely determined. One example of a well-established unique kinematic mode is in the plane-strain problem. In the plane-strain condition, unless one family of the characteristics is straight, the governing equation of the velocity field becomes the telegraphy equation which is hyperbolic and, therefore, the solution is uniquely determined if the boundary curve is not along a characteristic.

It can be readily shown that under certain boundary conditions the set $(\sigma_{ij}^{(1)}, C\dot{\varepsilon}_{ij}^{(1)})$ also satisfies the boundary conditions on S_v and therefore the deformation mode is clearly not unique. The following is a partial list of such boundary conditions.

(1) $S_v = 0$, i.e., all the boundaries are traction boundaries;

(2) $\hat{v}_i = 0$ on S_v;

(3) On S_v only the ratio between the velocity components are prescribed, e.g., $\frac{\hat{v}_i}{\hat{v}_j} = \alpha$;

(4) Mixed boundary condition; e.g., a normal component of $\hat{v}_i$ and a tangential component of $\hat{T}_i$ are prescribed over the surface, or vice versa. In this case, the additional condition of whether all the characteristics meet on a curve in the region should be checked [10].

Concrete examples are (1) the expansion of spherical shells [11] or cylindrical shells [12] under internal pressure, and (4) the indentation of a semi-infinite body by a flat punch under the plane-strain condition [13], torsion of a prismatic bar [10]. Among sheet-metal forming processes, hydrostatic bulging belongs to case (2) and punch stretching to case (4) or (3).

Note that the physical meaning of these boundary conditions is that the plastic flow is unconstrained and all or part of the body is free to deform. Mathematically, this nonuniqueness is due to the fact that the Levy-Mises theory, implied in the variational formulation π_1 and also appearing in the differential equations (2c), does not include the "viscosity effect" (in Prager's terminology [9]) and, therefore, this indeterminacy would be resolved if the workhardening effect is taken into account. In fact, for the workhardening solid there is no inherent indeterminacy in general; the apparent nonuniqueness is due simply to an inadequate formulation of the problem. In proper formulation, traction rate $\dot{T}_i$ must be specified on S_T, and then from an infinite number of kinematically possible modes the actual mode can be singled out by the additional requirement that there must exist an equilibrium distribution of stress rate compatible with the implied rate of hardening everywhere in the body and with the given traction rate $\dot{T}_i$ on S_T. Besides, the workhardening effect is explicitly brought into the constitutive equation in the form of

$$h\dot{\varepsilon}_{ij} = \frac{\sigma'_{ij}}{\bar{\sigma}}\dot{\bar{\sigma}} \qquad (3)$$

where $\dot{\bar{\sigma}}$ is the time rate of $\bar{\sigma}$, h the workhardening effect of the material being equal to $\frac{2}{3}\frac{d\bar{\sigma}}{d\bar{\varepsilon}}$. It can be shown that the constitute equation (3) can always be reduced to the constitutive equation (2c), but not necessarily vice versa. Therefore, for a perfectly plastic solid, specifying the traction rate does not resolve the indeterminacy. Hill, then, showed that among all variational modes compatible with the boundary conditions for $\hat{v}_i$ on S_v and the existing stress distribution σ_{ij}, the actual mode minimizes the following expression when geometry changes are neglected (Hill [8]):

$$\pi_2 = \frac{1}{2}\int h(\dot{\varepsilon}^*_{ij})^2\, dv - \int_{S_T} \dot{T}_i v_i\, dS. \tag{4}$$

Note that the virtual mode $\dot{\varepsilon}^*_{ij}$ in π_2 should be normal to the yield surface at the existing stress point in the stress space due to the compatibility requirement with existing stress distributions. For statically indeterminate problems, however, there is a coupling between stress field and strain rate field and we have to solve these two sets of variables simultaneously.

2. <u>Geometry change</u>

When the effect of geometry change cannot be neglected during deformation, it is necessary to reconsider the specification of the loading on S_T and the stresses since the changes in shape and area of surface elements are themselves unknown.

Let X_i be the position vector in a Cartesian reference frame at time t and after an infinitesimal time δt, x_i be the position. Let us call the configuration at time t undeformed configuration and the one at time t + δt deformed configuration. When an actual force dP_i acts upon the area element da at time t + δt, there are various ways of reckoning this force.

First, the actual force dP_i is referred to the deformed configuration, or

$$dP_i = n_j \sigma_{ij} da, \tag{5a}$$

where n_j is the unit normal vector to the surface element of area da in a deformed configuration. The stress tensor σ_{ij} defined in this manner is called Cauchy stress tensor, or sometimes, true stress tensor.

Second, the actual force dP_i is referred to the undeformed configuration, or

$$dP_i = N_j S_{ij} dA, \tag{5b}$$

where N_j is the unit normal vector to the surface element of area dA in an undeformed configuration. The stress tensor S_{ij} defined in this manner is called the first kind of Kirchhoff stress tensor, or sometimes, nominal stress tensor. This tensor has the disadvantage of not being symmetric and therefore awkward to use in a constitutive equation with a symmetric strain tensor. Nonetheless, sometimes this stress tensor is used with nonsymmetric velocity gradients [14].

Third, to obtain a stress tensor, which is symmetric and referred to the undeformed configuration, we proceed as follows. Instead of the actual force dP_i, consider a force $d\tilde{P}_i$ related to the force dP_0 in the same way that a material vector dX_i is related by the deformation to the corresponding vector dx_i. That is,

$$d\tilde{P}_i = \frac{\partial X_i}{\partial x_j} dP_j . \tag{5c}$$

Refer this pseudo-force $d\tilde{P}_i$ to the undeformed configuration to define the second kind of Kirchhoff stress tensor τ_{ij}:

$$d\tilde{P}_i = N_j \tau_{ij} dA. \tag{5d}$$

Using the expression relating the area change of the same element during deformation [15],

$$n_i da = \frac{\rho_0}{\rho} N_j \frac{\partial X_j}{\partial x_i} dA, \tag{6}$$

where ρ_0 and ρ are densities of the volume element before and after the deformation, the relationship between different stress measures is obtained. From Eqs. (6), (5a), and (5b),

$$\begin{aligned} dP_i = n_j \sigma_{ij} da &= \sigma_{ij} \frac{\rho_0}{\rho} N_k \frac{\partial X_k}{\partial x_j} dA \\ &= N_j S_{ij} dA \end{aligned} \tag{7a}$$

or

$$S_{ij} = \frac{\rho_0}{\rho} \frac{\partial X_j}{\partial x_k} \sigma_{ik} \tag{7b}$$

and from Eqs. (6), (5a), (5c), and (5d),

$$\tau_{ij} = \frac{\rho_0}{\rho} \frac{\partial X_i}{\partial x_k} \sigma_{kr} \frac{\partial X_j}{\partial x_r}. \tag{7c}$$

All these different stress tensors become exactly the same when we bring the deformed configurations to the undeformed configurations and make them identical in the limit. Stress rates, however, are not the same. Let δu_i be the increment of displacement of the element; then

$$\delta S_{ij} = \delta\sigma_{ij} - \sigma_{kj} \frac{\partial}{\partial x_k} \delta u_i \quad \text{to the first order,}$$

neglecting plastic volume change. Or, in terms of rates,

$$\dot{S}_{ij} = \dot{\sigma}_{ij} - \sigma_{kj} v_{i,k} \, . \tag{8}$$

Let us compare the magnitude of the second term with that of the first term in the right-hand side of Eq. (8). Since

$$\frac{\partial v_i}{\partial x_k} = \frac{1}{2}\left(\frac{\partial v_i}{\partial x_k} + \frac{\partial v_k}{\partial x_i}\right) + \frac{1}{2}\left(\frac{\partial v_i}{\partial x_k} - \frac{\partial v_k}{\partial x_i}\right)$$

$$= \frac{1}{2}\dot{\varepsilon}_{ik} + \frac{1}{2}\dot{\omega}_{ik}, \tag{9a}$$

where $\dot{\varepsilon}_{ik}$ is the rate of deformation and $\dot{\omega}_{ik}$ the rate of rotation. Then

$$\sigma_{kj}\frac{\partial v_i}{\partial x_k} = \frac{1}{2}\sigma_{kj}\dot{\varepsilon}_{ik} + \frac{1}{2}\sigma_{kj}\dot{\omega}_{ik}. \tag{9b}$$

Now, if workhardening characteristics are given by the relation

$$\bar{\sigma} = H(\bar{\varepsilon}), \tag{10a}$$

then

$$H' = \frac{d\bar{\sigma}}{d\bar{\varepsilon}}$$

or

$$d\bar{\varepsilon} = \frac{d\bar{\sigma}}{H'}\ . \tag{10b}$$

For an order of magnitude calculation we can write approximately

$$\dot{\varepsilon}_{ik} = \frac{\dot{\sigma}_{ik}}{H'} \tag{10c}$$

or

$$\sigma_{kj}\dot{\varepsilon}_{ik} = \frac{\sigma_{kj}\dot{\sigma}_{ik}}{H'}\ . \tag{10d}$$

Then, from Eqs. (9b) and (10d), Eq. (8) becomes

$$\dot{S}_{ij} = \dot{\sigma}_{ij}\left(1 - \frac{\sigma_{ij}}{H'}\right) + \frac{1}{2}\sigma_{kj}\dot{\omega}_{ik}. \tag{11}$$

From Eq. (11) we conclude that if the order of the rate of rotation $\dot{\omega}_{ij}$ is the same as or less than the order of strain rate $\dot{\varepsilon}_{ij}$, and if the work-hardening rate H' is greater than the stress level, then $\dot{S}_{ij} = \dot{\sigma}_{ij}$. Otherwise, geometry change should not be neglected.

It could be shown [16] that when geometry change is taken into account, the condition for continuing equilibrium requires that

$$\frac{\partial S_{ij}}{\partial x_i} = 0 \quad \text{in the absence of body force.}$$

Using this condition, Hill subsequently derived the following variational formulation [17]:

$$\pi_3 = \frac{1}{2}\int h(\dot{\varepsilon}^*_{ij})^2\, dv - \frac{1}{2}\int \sigma_{kj} v^*_{i,k} v^*_{j,i}\, dv - \int_{S_T} T_i v^*_i\, dS. \tag{12}$$

Formulation π_3 follows essentially the same line of formulation π_2 except that now geometry change is considered. In the formulation π_3, as well as in π_2, virtual mode must be compatible with the existing stress distribution and the boundary condition on S_v. As has been discussed earlier for statically indeterminate problems this is not an appropriate formulation.

Summarizing the development so far, the kinematic mode in sheet-metal forming of a rigid-plastic solid is not uniquely determined by considering the first-order expansion of the potential alone. Consideration up to second-order expansion of the potential, or equivalent consideration of workhardening rate in a physical sense, needs stress rate terms explicitly in the variational formulation. When geometry change cannot be neglected, these stress rate- are related to stress distribution, which is not known for statically indeterminate problems. The approach of viewing the deformation as determining the incipient flow by assuming the deformed configuration coincident with the undeformed configuration clearly does not lead to a workable variational formulation for sheet-metal forming of a rigid-plastic solid. In this respect, it is intended to develop an appropriate variational formulation in the next section.

SECTION III

FINITE-ELEMENT FORMULATION

1. Variational formulation

Let x_i be the position vector in a Cartesian frame of reference at time t, the moment under consideration. Let σ_{ij} be the true stress at time t and $\sigma_{ij} + d\sigma_{ij}$ the true stress in the same material element after an infinitesimal time dt, both tensors being associated with the same Cartesian axes. Let ds_{ij} be the increment in nominal stress in the same element in time dt, based on the dimensions at time t. Let du_i be the increment of displacement of the element, then

$$ds_{ij} = d\sigma_{ij} - \sigma_{j} \frac{\partial(du_i)}{\partial x_k} \; . \tag{13}$$

Requiring continuing equilibrium of stresses, the virtual work principle gives

$$\int_V \left(\sigma_{ij} + d\sigma_{ij} - \sigma_{kj} \frac{\partial(du_i)}{\partial x_k}\right) \delta\left(\frac{\partial(du_j)}{\partial x_i}\right) dV = \int_{S_F} (E_j + dF_j)\delta(du_j)\, ds, \tag{14}$$

where $T_j = \ell_i \sigma_{ij}$ and $dT_j = \ell_i ds_{ij}$, ℓ_i being the unit normal to the surface at time t. The variational formulation is obtained from Eq. (14) as follows:

$$\delta\phi = \delta\left\{\int_V \sigma_{ij} d\varepsilon_{ij}\, dV + \int_V \frac{1}{2} h d\varepsilon_{ij}^2\, dV - \int_V \frac{1}{2} \sigma_{kj} \frac{\partial(du_i)}{\partial x_k} \frac{\partial(du_j)}{\partial x_i}\, dV \right.$$

$$\left. - \int_{S_F} (T_j + dT_j) du_j\, ds\right\} = 0,$$

where

$$d\varepsilon_{ij} = \frac{1}{2}\left(\frac{\partial(du_i)}{\partial x_j} + \frac{\partial(du_j)}{\partial x_i}\right) \tag{15}$$

and $h = \frac{2}{3} H'$, with H' the slope of the stress and strain curve. The first three terms of the functional ϕ represent the energy dissipated during the time dt up to the second order. If it is assumed that the principal axes of true strain-rate keep the same directions in the element and the principal components of strain-rate maintain the constant ratios during the time dt, the dissipated energy can be expressed directly [18] as

$$\sum (\sigma_p dE_p + \frac{1}{2} h dE_p^2)$$

per unit volume, where dE_p is the logarithmic strain components. The final form of the functional becomes

$$\phi = \int_V \bar{\sigma} d\bar{E}\ dV + \frac{1}{2}\int H'(d\bar{E})^2\ dV - \int_{S_F} (T_j + dT_j)\ du_j\ dS, \tag{16}$$

where $d\bar{E}$ is defined by

$$d\bar{E} = \sqrt{\frac{2}{3}\sum (dE_p)^2}$$

2. Theory of the finite-element method

An important step in finite-element modeling is obtaining approximate state equations in a region. The weighted residual method derives the state equations directly from the governing differential equations. Let us write the governing differential equation as

$$Lu - f = 0, \tag{17}$$

where L is the differential operator, f is the known function, and u is the solution. With the trial solution u*, Eq. (17) is not satisfied, but there remains an error or residual R such that

$$R = Lu^* - f. \tag{18}$$

This residual is multiplied by weight function w and integrated over the domain and the state equations are derived from the condition that this integral vanishes with a given choice of weight function w:

$$\int wR \, dv = 0. \tag{19}$$

One well-known method among weighted residual methods is Galerkin's approach.

A more frequently used approach is the derivation from a variational principle which is a dual expression of the governing differential equation. Assume that a functional Φ, which is equivalent to the differential equation, has been established. Let a continuum be divided into a finite collection M of subdomains called elements interconnected at a finite number of nodes N. If it is true that the total functional is equal to the sum of the contributions of each element $\phi^{(m)}$, then we may write as follows:

$$\Phi = \sum_{m=1}^{M} \phi^{(m)}(u). \tag{20}$$

In each element let us approximate the solution with a linear combination of trial functions v_i such that

$$u \simeq \sum \alpha_i v_i \tag{21}$$

holds, where α_i are unknown coefficients to be determined later. By substituting Eq. (21) into Eq. (20), we have

$$\Phi = \sum_{m=1}^{M} \varphi^{(m)}(\alpha_i v_i)$$

$$= \sum_{m=1}^{M} \varphi^{(m)}(\alpha_i) \quad \text{since } v_i\text{'s are known} \tag{22}$$

$$= \varphi(\alpha_i).$$

The original Φ of u is now discretized with a function φ of parameters α_i, and the initial variational problem reduces to determining the α_i that minimizes φ. The minimization of φ with respect to α_i may be written as

$$\delta\varphi = \frac{\delta\varphi}{\partial\alpha_i}\,\delta\alpha_i = 0, \tag{23}$$

where δ denotes the first variation. Since α_i's are independent, expression (23) is equivalent to a set of simultaneous equations,

$$\frac{\partial\varphi}{\partial\alpha_i} = 0. \tag{24}$$

This is, in fact, the classical Ritz technique. It is the choice of trial functions that makes the finite-element method different from the Ritz method and renders it successful; they are piecewise polynomials. Bsides, the coefficients α_i, called nodal values in the finite-element literature, do have a definite physical meaning, such as displacement or velocity.

The trial function v_i must satisfy certain requirements to enable convergence as the subdivision into ever smaller elements is attempted. First, as the element size decreases, the functions in the integral must tend to be single-valued and well behaved in physical problems. Thisis called the "completeness" requirement and is satisfied if the trial function is of class c^p when p is the highest order in the integrand of the functional. Second, the validity of the summation implied in Eq. (20) must be preserved. This is called the "compatibility" requirement and is satisfied if v_i is of class C^{p-1} [19], [20]. When admissible trial functions are used, the functional converges monotonically with an increasing number of elements (or decreasing size) at a rate proportional to h^2 where h is a characteristic element dimension.

3. Modeling of axisymmetric problems

The general outline of the finite-element modeling stated above will be expanded in detail for the case of axisymmetric thin shells subject to axisymmetric loading. This particular problem is of interest since some basic sheet-metal forming processes belong to this category. When the ratio of thickness to the radius of curvature is sufficiently small, bending moment and shearing forces may be neglected without serious error and the *membrane theory may* be justified [21]. Moreover, the state of stress can be treated as an approximate plane so long as $\frac{dt}{ds}$ is small compared with unity, where t is the local thickness and s is the distance in any direction parallel to the surface. We now may rewrite Φ with the substitution of $t\ dA = dv$ to Eq. (16):

$$\Phi = \int \bar{\sigma}(d\bar{E})t\ dA + \frac{1}{2}\int H'(d\bar{E})^2(t\ dA) - \int (T + dT)du_j\ dA \tag{25}$$

for the unit included angle of the element, where A is the area of the element and t is the sheet thickness.

From the symmetry of the problem it is easily shown that the circumferential direction and the meridian direction are the principal directions and if the friction between the shell and the external agent is negligible, the thickness direction will be the third principal direction. Within the order of approximation taken in the formulation, the logarithmic strain increment may be used as the strain increment measure. Then the definitions of strain increments are

$$dE = \begin{Bmatrix} dE_1 \\ dE_2 \end{Bmatrix} = \begin{Bmatrix} dE_r \\ dE_\theta \end{Bmatrix} = \begin{Bmatrix} \ln \dfrac{s}{s_0} \\ \ln \dfrac{r}{r_0} \end{Bmatrix} , \tag{26}$$

if, during an incremental deformation, an element of undeformed length s_0 is stretched to the length s and the point currently at the radial distance r_0 moves to the deformed radial location r. Subscripts r, θ refer to the meridian and the circumferential direction, respectively.

To bring the model closer to reality in the present investigation, normal anisotropy is included and the corresponding stress-strain increment relation is obtained, using Hill's criterion [13], as

$$\frac{dE_r}{(1 + R)\sigma_r - R\sigma_\theta} = \frac{dE_\theta}{(1 + R)\sigma_\theta - R\sigma_r} = \frac{d\bar{E}}{(1 + R)\bar{\sigma}} , \tag{27}$$

where R is the planar isotropy parameter which is the ratio of width strain to the thickness strain in uniaxial tension. The effective stress and the effective strain are defined† as

$$\bar{\sigma} = \sqrt{\sigma_\theta^2 - \frac{2R}{1 + R}\sigma_r\sigma_\theta + \sigma_r^2} , \tag{28a}$$

$$d\bar{E} = \frac{1 + R}{\sqrt{1 + 2R}} \sqrt{dE_r^2 + \frac{2R}{1 + R} dE_\theta dE_r + dE_\theta^2} . \tag{28b}$$

---

†Note that $H' = \dfrac{d\bar{\sigma}}{d\bar{E}}$ must be consistent with these definitions.

The effective strain, $d\bar{E}$, may be written in matrix form as

$$d\bar{E} = \sqrt{\frac{2}{3}}\,[d\underset{\sim}{E}^T \underline{D}\; dE]^{1/2}, \tag{29a}$$

where

$$D = \frac{3(1 + R)}{2(1 + 2R)} \begin{vmatrix} 1 + R & R \\ R & 1 + R \end{vmatrix}. \tag{29b}$$

The sheet geometry is approximated by a series of conical frustra, as shown in Fig. 1. Linear trial functions, or shape functions, as they are often called in the finite-element literature, are enough since the integrand in the functional is of class C^1. The unknown coefficients, or nodal values, are taken to be the incremental displacement at nodes. Then we may write

$$\begin{aligned} \underset{\sim}{u}^{(m)} &= \langle dv_1, dw_1, dv_2, dw_2 \rangle^T \\ &= \langle du_1, du_2, du_3, du_4 \rangle^T \end{aligned} \tag{30}$$

for a representative element m, where dv_i, dw_i are the radial and the axial components of incremental displacement of the i-th node. Then the incremental displacement field inside the element may be written as

$$\begin{aligned} \underset{\sim}{u} = \begin{Bmatrix} d\underset{\sim}{v} \\ d\underset{\sim}{w} \end{Bmatrix} &= \begin{vmatrix} \frac{1 + t'}{2} & 0 & \frac{1 - t'}{2} & 0 \\ 0 & \frac{1 + t'}{2} & 0 & \frac{1 - t'}{2} \end{vmatrix} \begin{Bmatrix} dv_1 \\ dw_1 \\ dv_2 \\ dw_2 \end{Bmatrix} \\ &= \underline{N}\underset{\sim}{u}^{(m)}, \end{aligned} \tag{31}$$

where t' is the local coordinate varying from the value of -1 at node 2

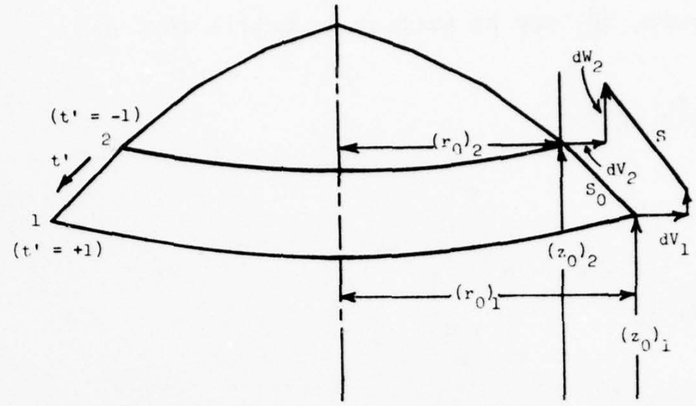


Figure 1. Approximation of the Sheet Geometry into a Series of Conical Frustra

to +1 at node 1. (See Fig. 1.) Due to this incremental displacement field, an element of length s_0,

$$s_0 = \sqrt{\{(r_0)_1 - (r_0)_2\}^2 + \{(z_0)_2 - (z_0)_1\}^2},$$

is stretched to a new length s,

$$\begin{aligned} s &= \sqrt{\{(r_0)_1 - (r_0)_2 + dv_1 - dv_2\}^2 + \{(z_0)_2 - (z_0)_1 + dw_2 - dw_1\}^2} \\ &= \sqrt{(r_1 - r_2)^2 + (z_2 - z_1)^2}, \end{aligned} \tag{32}$$

where $(r_0)_i$, $(z_0)_i$ are the radial and the vertical positions of the i-th node at the undeformed configuration and $(r)_i$, $(z)_i$ at the deformed configuration. Since the element is straight, any point of t' in the local coordinate is shown to have a global radial position r_0 determined by

$$r_0 = (\frac{1 + t'}{2})(r_0)_1 + (\frac{1 - t'}{2})(r_0)_2. \tag{33}$$

The new position r of the same particle is given by

$$r = r_0 + \frac{(1 + t')}{2} dv_1 + \frac{(1 - t')}{2} dv_2. \tag{34}$$

We are now at the position of calculating the strain increment field. Recall the equation (25) and substitute Eqs. (32), (33), and (34) into it to obtain

$$
d\underset{\sim}{E} = \begin{Bmatrix} \dfrac{1}{2} \ln \dfrac{\{(r_0)_1 - (r_0)_2 + dv_1 - dv_2\}^2 + \{(z_0)_1 - (z_0)_2 + dw_2 - dw_1\}^2}{s_0^2} \\ \ln \dfrac{r_o + \dfrac{(1 + t')}{2} dv_1 + \dfrac{(1 - t')}{2} dv_2}{r_0} \end{Bmatrix} \tag{35}
$$

We may write $\phi^{(m)}$, a contribution from the m-th element to the total functional Φ, in terms of nodal values, for unit angle included:

$$
\begin{aligned}
\phi^{(m)} &= \int \{\bar{\sigma} d\bar{E} + \frac{1}{2} H'(d\bar{E})^2 \, t \, dA - \int (T_i + dT_i) v_i \, dA \\
&= \int \bar{\sigma}(\sqrt{\tfrac{2}{3}} \, t) [d\underset{\sim}{E}^T \underline{D} \, d\underset{\sim}{E}]^{1/2} \, dA + \frac{1}{2} \int H'(\tfrac{2}{3} t) [d\underset{\sim}{E}^T \underline{D} \, d\underset{\sim}{E}] \, dA - \int \underset{\sim}{T}^T \underline{N} \underset{\sim}{u}^{(m)} \, dA \, ,
\end{aligned} \tag{36}
$$

where

$$
T = \begin{Bmatrix} T_1 + dT_1 \\ T_2 + dT_2 \\ T_3 + dT_3 \\ T_4 + dT_4 \end{Bmatrix} .
$$

Minimization gives a set of simultaneous equations:

$$
\begin{aligned}
\frac{\partial \phi^{(m)}}{\partial \underset{\sim}{u}^{(m)}} &= \int (\sqrt{\tfrac{2}{3}} \, t) \bar{\sigma} [d\underset{\sim}{E}^T \underline{D} \, d\underset{\sim}{E}]^{-1/2} \frac{\partial (d\underset{\sim}{E})^T}{\partial \underset{\sim}{u}^{(m)}} \underline{D} \, d\underset{\sim}{E} \, dA + \int (\tfrac{2}{3} t) H' \frac{\partial (d\underset{\sim}{E})^T}{\partial \underset{\sim}{u}^{(m)}} \underline{D} \, d\underset{\sim}{E} \, dA \\
&\quad - \int \underline{N}^T \underset{\sim}{T} \, dA \, .
\end{aligned} \tag{37}
$$

From Eq. (35),

$$
\frac{\partial (d\underset{\sim}{E})}{\partial \underset{\sim}{u}^{(m)}} = \underline{Q} = \begin{vmatrix} \dfrac{\partial(dE_1)}{\partial u_1} & \dfrac{\partial(dE_2)}{\partial u_1} \\ \dfrac{\partial(dE_1)}{\partial u_2} & \dfrac{\partial(dE_2)}{\partial u_2} \\ \dfrac{\partial(dE_1)}{\partial u_3} & \dfrac{\partial(dE_2)}{\partial u_3} \\ \dfrac{\partial(dE_1)}{\partial u_4} & \dfrac{\partial(dE_2)}{\partial u_4} \end{vmatrix} = \begin{vmatrix} \dfrac{r_1 - r_2}{s^2} & \dfrac{1 + t'}{2r} \\ \dfrac{-(z_2 - z_1)}{s^2} & 0 \\ \dfrac{-(r_1 - r_2)}{s^2} & \dfrac{1 - t'}{2r} \\ \dfrac{(z_2 - z_1)}{s^2} & 0 \end{vmatrix} \tag{38}
$$

Therefore, Eq. (37) becomes

$$
\frac{\partial \phi^{(m)}}{\partial \underset{\sim}{u}^{(m)}} = \int (\tfrac{2}{3}\, t)\bar{\sigma}[\tfrac{2}{3}\, d\underset{\sim}{E}^T \underline{D}\; d\underset{\sim}{E}]^{-1/2} \underline{QD}\; d\underset{\sim}{E}\; dA + \int (\tfrac{2}{3}\, t) H' \underline{QD}\; d\underset{\sim}{E}\; dA - \int N^T \underset{\sim}{T}\; dA = 0. \tag{39}
$$

These equations, being valid for an m-th element, are now to be combined under the condition of compatibility that the first-order derivative of nodal value may be discontinuous across element boundaries but the nodal value itself must be continuous,

$$
\frac{\partial \Phi}{\partial \underset{\sim}{u}} = \sum \frac{\partial \phi^{(m)}}{\partial \underset{\sim}{u}^{(m)}} = 0. \tag{40}
$$

4. Linearization

Eqs. (39) and (40) are nonlinear equations and it is very difficult to solve them without linearizing. One way is to take an initial guess of the solution to the equation as u* and rewrite Eq. (39) in terms of the differences between this initial guess and the correct solution Δu, where $u_{correct} = u^* + \Delta u$, and expand it. Where the initial guess is sufficiently close to the correct solution, we may neglect higher-order terms of Δu and thereby linearize successfully. This can be done mathematically in a systematic way and is called the Newton-Raphson method [22]. Say we have a nonlinear equation $\psi(u) = 0$, then we may expand into a series with respect to the correct solution u_0 such that

$$\psi(u) = \psi(u_0) + \left(\frac{d\psi}{du}\right)_{u=u_0}(u - u_0) + \frac{1}{2}\left(\frac{d^2\psi}{du^2}\right)_{u=u_0}(u - u_0)^2 + \cdots$$

$$= \psi^* + \left(\frac{d\psi}{du}\right)^*\Delta u + \frac{1}{2}\left(\frac{d^2\psi}{du^2}\right)(\Delta u)^2 + \cdots = 0.$$

If u and u_0 are sufficiently close, we may neglect the higher-order terms and write

$$\psi = \psi^* + \left(\frac{d\psi}{du}\right)^*\Delta u = 0. \tag{41}$$

In our formulations the equations to be minimized are $\frac{\partial\varphi^{(m)}}{\partial \underset{\sim}{u}^{(m)}} = 0$, and, therefore, the expressions corresponding to Eq. (41) are

$$\left|\frac{\partial^2\varphi^{(m)}}{\partial u_i^{(m)}\partial u_j^{(m)}}\right|^*(\Delta u) = \left|-\frac{\partial\varphi^{(m)}}{\partial u_i}\right|^*. \tag{42}$$

It may be shown that

$$\frac{\partial^2\varphi^{(m)}}{\partial u_i^{(m)}\partial u_j^{(m)}} = P^{(m)} = \frac{2}{3}\int \frac{1}{d\bar{E}}\left\{\left(\bar{\sigma} + H'\ d\bar{E})(K - \frac{2}{3}\frac{bb^T}{d\bar{E}}\right) + \frac{2}{3}\frac{H'bb^T}{d\bar{E}^2}\right\}t\ dA \tag{43a}$$

where

$\underset{\sim}{b} = QD\ d\underset{\sim}{E}$

$\underline{K} = QDQ^T$

and that

$$\frac{\partial \varphi^{(m)}}{\partial u_i^{(m)}} = \underset{\sim}{H}^{(m)} - \underset{\sim}{F}^{(m)}, \tag{43b}$$

where

$$H^{(m)} = \frac{2}{3} \int \frac{1}{d\bar{E}} (\bar{\sigma} + H'\overline{d\bar{E}}) bt \; dA,$$

$$F^{(m)} = \int \underline{N}^T \underset{\sim}{T} \; dA.$$

By assembling the equations obtained for an element, we finally have

$$P^* \Delta u = F - H^* \tag{44}$$

We evaluate the integrals with the Gaussian quadrature formulation.

We have yet to introduce the boundary conditions for solving a physical problem. For an incremental displacement prescribed boundary, the corresponding perturbations should vanish and, for a traction prescribed boundary, the prescribed traction value will enter into the $\underline{F}$ vector. The solution procedure is as follows:

(1) Assume an initial guess u_1, and compute P, H, F corresponding to this guess.

(2) Solve Eq. (3.31) and obtain Δu.

(3) Obtain a new initial guess $u_2 = u_1 + \Delta u$.

Repeat this process until convergence is achieved. Convergence is checked by the fractional norm. A norm is defined by a square root value, i.e.,

$$\|u\| = \sqrt{u_1^2 + u_2^2 + \cdots}$$

and

$$\|\Delta u\| = \sqrt{(\Delta u_1)^2 + (\Delta u_2)^2 + \cdots}$$

The fractional norm is the ratio $\frac{\|\Delta u\|}{\|u\|}$ and when, for subsequent iterations, this value reaches the magnitude smaller than a predetermined value, say, 10^{-6}, the iteration stops and the solution is thus obtained.

SECTION IV

HYDROSTATIC BULGING

1. Introduction

The ductility of sheet metal under biaxial stress is often examined by means of the so-called bulge test. A uniform plane sheet is placed over a die with an aperture and is firmly clamped around the perimeter. An increasing hydrostatic pressure is applied to one side of the sheet, causing it to bulge through the aperture. From the measured profile and thickness of the plastically deformed sheet near the pole, it is possible to calculate the local state of stress in terms of the applied pressure. If, in addition, the state of strain is measured by means of a grid, the stress-strain characteristics of the metal under biaxial tension are obtained. The advantage of this test over any other simple one is that a greater range of pre-instability strain can be obtained.

Hydrostatic bulging is not only important as a material property test, but also as a forming operation. Thus, a number of theoretical investigations, dealing with axisymmetric hydrostatic bulging (Fig. 2) has appeared in the literature.

The classical analysis of bulging is the one by Hill [23]. His solutions are, however, special ones. Instead of analyzing deformation with a given stress-strain characteristic, Hill first adopted special kinematic assumptions and from them deducted the necessary stress-strain characteristics which satisfy all the governing equations under the prescribed kinematic mode. The kinematic assumptions are first, that any material element describes a circular path which is, moreover, orthogonal to the

Figure 2 Schematic view of hydrostatic bulging.

momentary profile, and second, that circumferential strain is numerically equal to the tangential strain. The required stress-strain characteristic is found to be an exponential type. Hill's other solution on a linear workhardening solid uses the method of successive approximation by adopting a yield criterion which is neither von Mises nor Tresca, for the purpose of mathematical simplicity.

Analyses of work by Woo [24], Yamada [25], and Wang [26] are based upon the realistic choices of stress-strain characteristics and the yield criterion. In applying the deformation theory of rigid plasticity, Wang experiences a mathematical difficulty and attributes this to the fact that the differential equations associated with the deformation theory possesses a singularity which has the effect of restricting the range of calculation within a certain value of the polar strain. Besides, the agreement of deformation theory predictions with the experiment is rather poorer than the incremental theory prediction [27].

In applying the incremental theory of rigid plasticity, researchers experience a difficulty in satisfying the boundary condition at the fixed edge, i.e., $\dot{\varepsilon}_\theta = 0$. To avoid this difficulty, Woo uses the deformation theory, while Yamada reasons that introducing an elastic strain component into the formulation will resolve this "mathematical difficulty" (in Yamada's terms) and turns to the elasto-plastic constitutive law. Another theoretical work of interest comes from Wang, using the parametric representation of the stresses.

The only published solution on hydrostatic bulging using the finite-element method is the one by Iseki et al. [28], with the incremental theory of elasto-plasticity.

2. Computational procedures

In adopting the finite-element model to hydrostatic bulging, it is necessary to reconsider the external work increment term, since the pressure is uniform over the entire surface of a closed shell. In this case the increment of external work may be written as [29], [30],

$$\Delta w = p\nabla\bar{V}, \tag{45}$$

where $\nabla\bar{V}$ is the increase of the volume enclosed by the deformed sheet and p is the pressure acting on the deformed configuration.

As an initial condition, Hill's special solution is utilized. In other words, the initial profile of the bulge is assumed to be a part of a sphere whose radius is given by $r = \frac{1}{2}\left(\frac{a^2}{h} + h\right)$, where a is the radius of the original blank and h is the polar height at the moment. With this geometry, a pressure p is prescribed. This pressure should be greater, at least, than the pressure which makes the sheet having initial geometry everywhere plastic. The initial guess on the incremental displacement is also obtained from Hill's special solution by assuming normal trajectory of the element particle to the bulge profile. The program for computing the initial guess is given in Appendix A.

When a converged solution is obtained for the given pressure, a new bulge profile is determined from the initial bulge profile and incremental displacement grid. Then the pressure is assigned a higher value and the converged solution for the previous step is used as the initial guess for the incremental displacement field and the computation continues in this way. The program for the analysis of hydraulic bulge is given in Appendix B.

3. Results and discussion

To examine the validity of the present FEM for hydrostatic bulging, the solution is compared with those achieved by the elasto-plastic FEM and the experiment.

The following conditions were employed for the comparison with the elasto-plastic FEM:

Workhardening characteristics: $\bar{\sigma} = 105(.0019 + \bar{\varepsilon})^{0.2} \times 10^{9}\mathrm{kg/m}^2$

$= 1.036(.0019 + \bar{\varepsilon})^{0.2} \times 10^{9}\mathrm{N/m}^2$

Thickness: 3.0×10^{-4}m (= 0.3 mm)

Radius of the sheet: 2.4×10^{-2}m (= 24 mm)

Anisotropy parameter: 1.0

An identical problem was also solved by Yamada [25], using the finite-difference method with the elastic-plastic theory. Fig. 3 shows the relationship between hydrostatic pressure and the polar thickness strain. The solid line represents the elasto-plastic FEM (and also the finite-difference method) and the points indicate the solution given by the rigid-plastic FEM. The deviation of the first point by the rigid-plastic FEM is thought to reflect the approximation involved in the initial condition that the sheet is everywhere plastic and that the initial geometry is a part of a sphere. The solution can be improved numerically by taking a smaller value of h in generating the initial condition. Nevertheless, the solutions after this first step are in extremely good agreement with the elasto-plastic FEM and any disturbance in the initial conditions does not matter after an initial deformation of a small magnitude. The pressure increment is raised by twice after some deformation and it is to be noted that the solutions

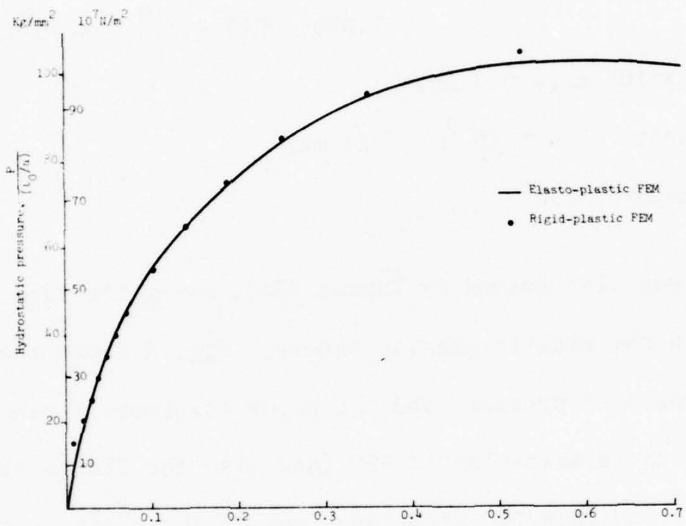


Figure 3. Hydrostatic pressure vs. polar thickness strain.

with the larger pressure increment size are still accurate. This means that the method is computationally economical with a reasonable accuracy. After the last point in the diagram the solution diverges and it is thought that the pressure maximum has been reached. The convergence is excellent; in every step, five to seven iterations seem to be sufficient. Fig. 4(a), (b) show the comparisons of strain distributions. The circumferential strain distributions are in good agreement. The tangential strain distribution by the rigid-plastic FEM deviates somewhat at the edge from that by the elasto-plastic FEM. The tangential strain is more sensitive to the method employed than the circumferential strain, but this deviation of tangential strain is not serious because the solution closely follows that by the finite-difference method and we may conclude that the strain distribution is accurately predicted. Fig. 5 shows the distributions of stresses when the polar thickness strain is (-0.4). Fig. 6 shows the bulge profile at some stages of deformation. A number of material elements are traced during deformation and are shown on each bulge profile.

Next, the solution is compared with Mellor's [31] experiment on half-hard aluminum.

Workhardening characteristics: $\sigma = 15{,}460(1 + 0.76\varepsilon)$ psi

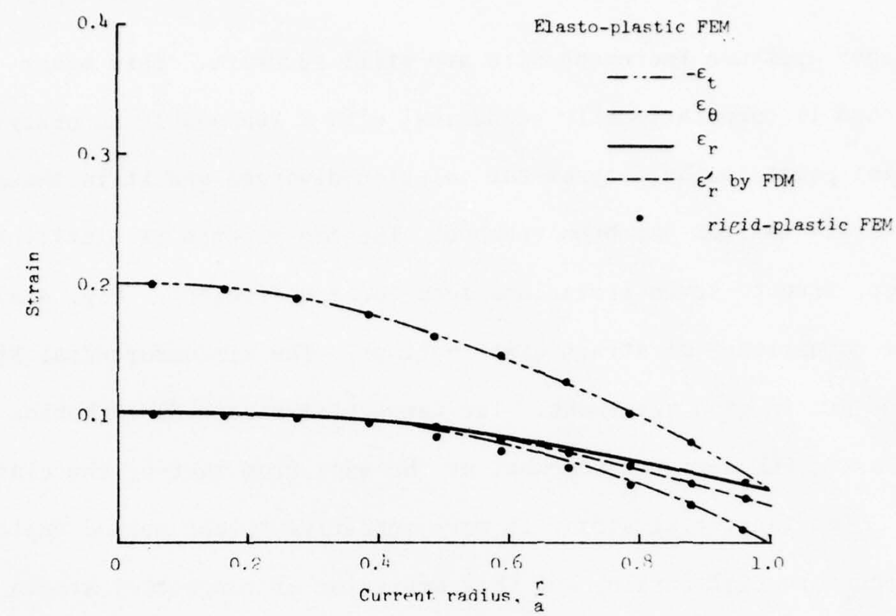
$$= 1.066(1 + 0.76\varepsilon) \times 10^8 \mathrm{N/m}^2$$

Radius of the sheet: 5.0 inches = 1.27 m

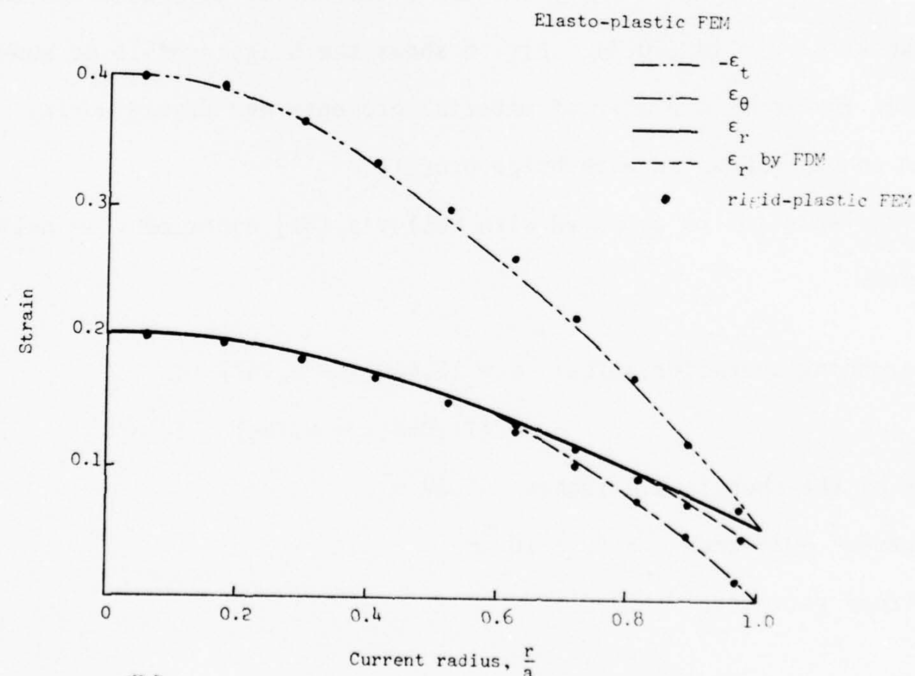
Thickness: .035 inch = 8.89×10^{-4}m

Anisotropy parameter: 1.0

One thing to be mentioned is that in the actual experiment, the die has a round profile of radius $\frac{3}{8}$ in., but in the analysis this profile has been



(a)



(b)

Figure 4. (a) and (b) Distribution of Strains.

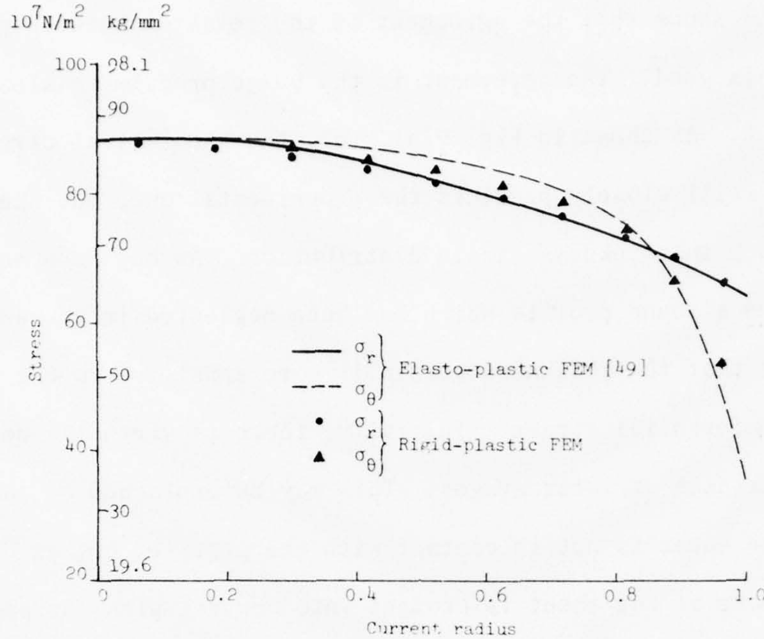


Figure 5. Distribution of Stresses

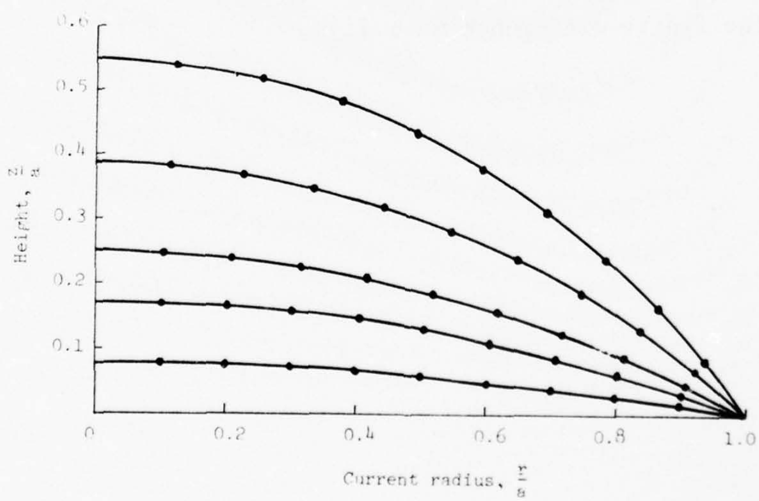


Figure 6. Bulge Profile

neglected. Fig. 7 shows that the agreement of the relation between pressure and polar height is good. The agreement in the bulge profile is also excellent, as in Fig. 8. As shown in Fig. 9(a), (b), the theoretical circumferential strain still closely predicts the experimental one, but there is some discrepancy in thickness strain distribution. As has been mentioned, the actual die has a round profile which has been neglected in the analysis, and it is thought that the thickness strain is more sensitive to the profile than is the circumferential strain. Initially, there is virtually no discrepancy, but increases at later stages. This may be explained by the fact that initially the sheet is not in contact with the profile, but as deformation continues, more of the sheet is brought into contact with the profile and makes the actual situation different from the one used in the analysis.

In general, the theoretical prediction by the rigid-plastic FEM is in good agreement with both the experiments and the analyses by the elasto-plastic FEM and the finite-difference method.

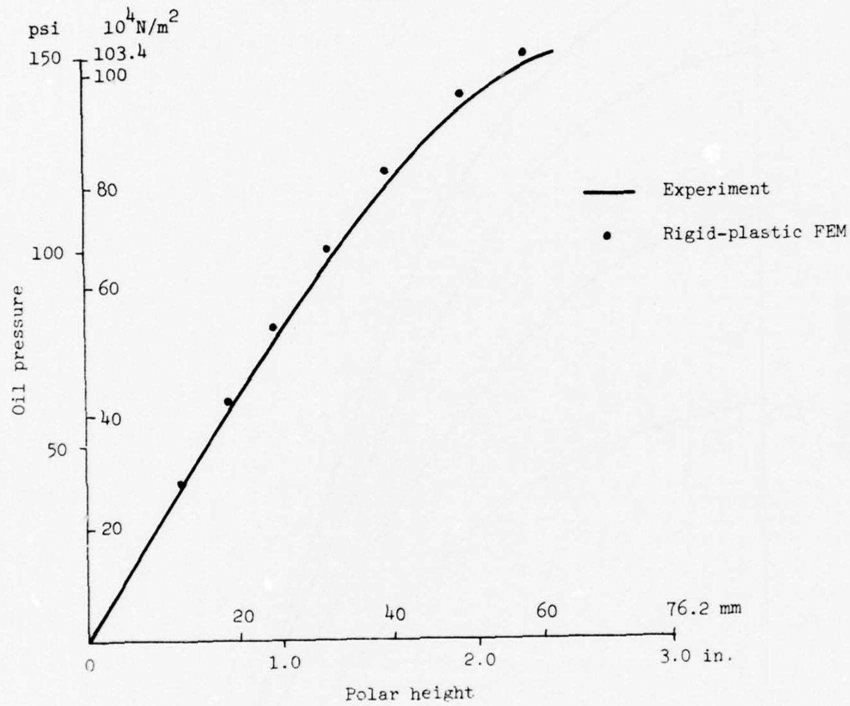


Figure 7. Polar Height vs. Pressure

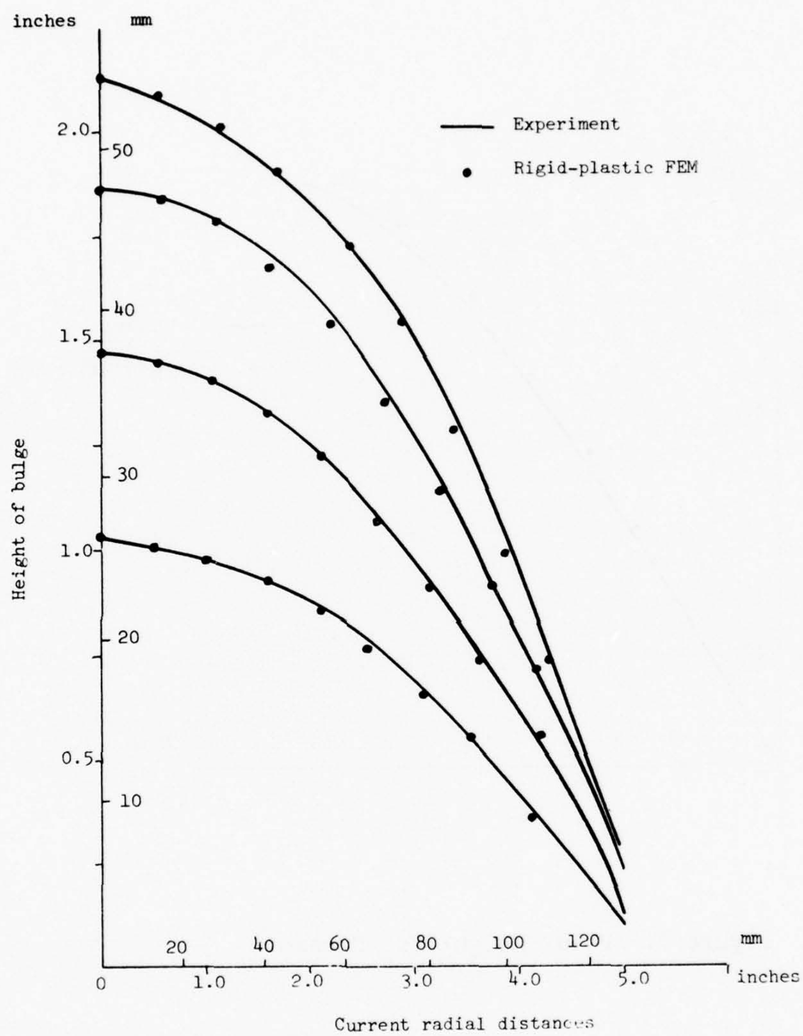


Figure 8. Bulge Profile

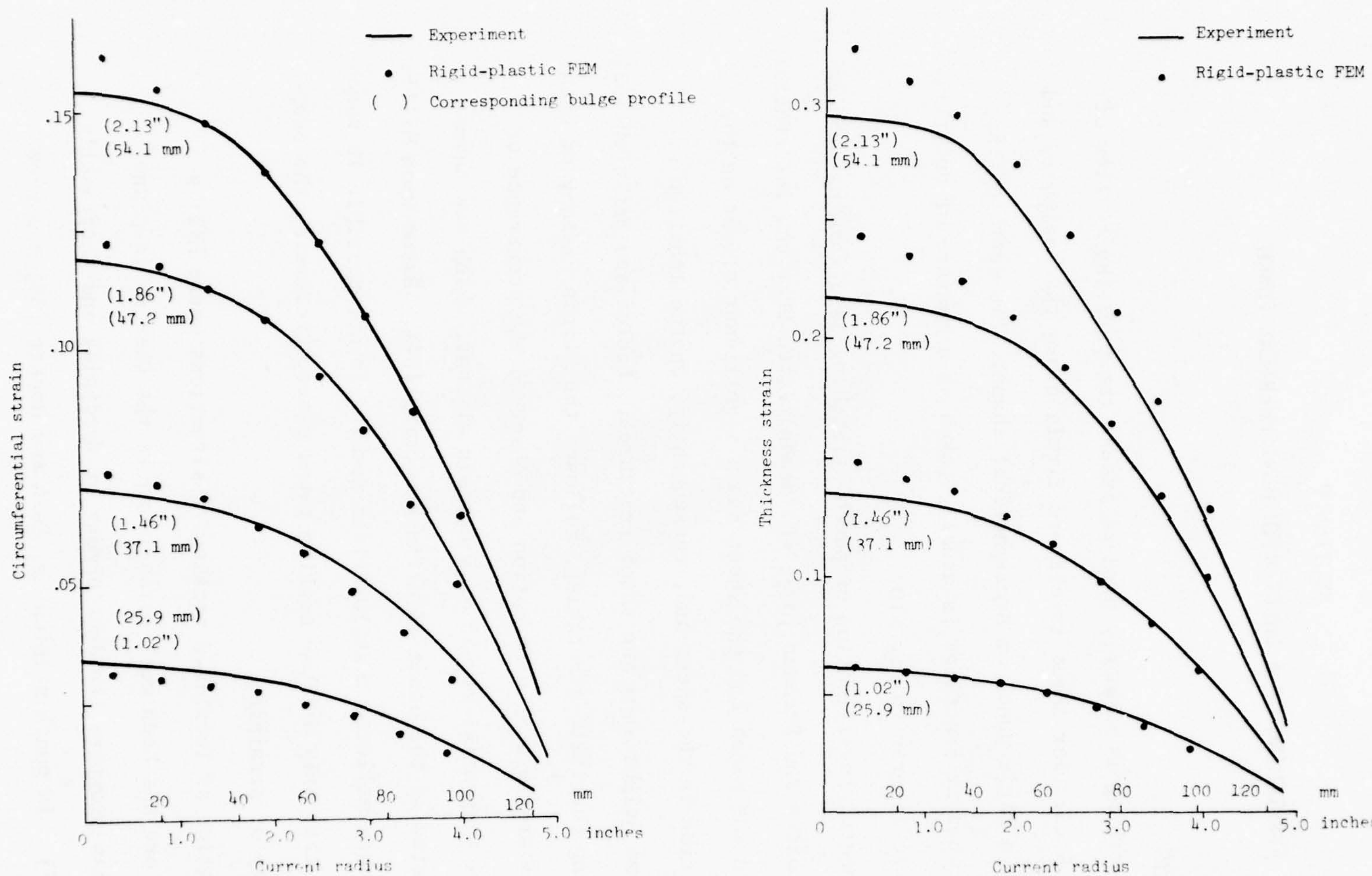


Figure 9. (a) Circumferential Strain Distribution; (b) Thickness Strain Distribution

SECTION V

STRETCHING OF A SHEET WITH HEMISPHERICAL PUNCH

1. Introduction

Punch stretching is commonly used to assess the pressing quality of sheet metals. A circular sheet is clamped firmly along the periphery and is stretched by a rigid punch of hemispherical shape. The depth of the deformed sheet when it fractures is usually taken as a measure of ductility. See the schematic diagram in Fig. 10.

An experimental investigation of punch stretching as a forming problem dates back to Loxely and Freeman [32], who demonstrated that the interfacial friction between the punch and the sheet has a significant effect on the strain distribution in the sheet and, consequently, on the location of fracture and dome height when the sheet fractures. Keeler and Backofen [33], in characterizing the limit stretching, followed the strain history of each element with the progress of deformation and observed the occurrence of discontinuity in tangential strain at a certain element, which was subsequently interpreted as the onset of diffuse necking [34]. Based upon Hill's analysis [35], they believed that localized necking is not possible in punch stretching, but that only diffuse necking takes place, increasing the overall nonuniformity of straining.

The observation of localized necking in situations where Hill's analysis denies one has been well established in the case of in-plane stretching and has prompted the development of Marciniak and Kuczynski's theory [36], [37]. In punch stretching, Gosh and Hecker [38] observed localized necking and reported that local necking sets in even though

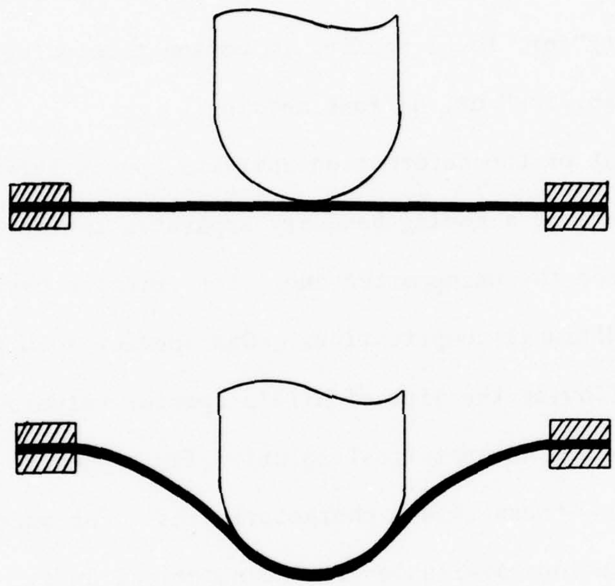


Figure 10. Schematic View of the Stretching of a Sheet with a Hemispherical Head Punch

the plane-strain condition, which is thought to be responsible for local necking in in-plane stretching, is not achieved. This is attributed to the fact that in punch stretching an increment in tangential strain is geometrically tied to an increment in circumferential strain and, therefore, the approach to plane-strain condition becomes slower. Another experimental investigation of punch stretching is the one by Alexander and Kaftanoglu [39]. They observed that the deformation is limited by the "strain propagation instability" or, local necking in common terminology, and not by "maximum load instability" or, diffuse necking.

From the viewpoint of the deformation analysis, punch stretching is a complicated problem because a moving boundary separates the region in contact with the punch head from the unsupported one. The friction over the punch head gives rise to additional complications. One special solution is by Chakrabarty [40]. Following the line of Hill's special solution on hydrostatic bulging he obtained an analytical solution for a special material having exponential type stress-strain characteristics. For more general materials the only solutions available are the numerical ones. Numerical solutions of importance are those by Woo [41] and by Wang [42], [43].

Woo's and Wang's solutions were obtained by the finite-difference method. The only solution by the finite-element method on punch stretching is one by Wifi [44]. His elasto-plastic, finite-element model does not neglect the bending moment nor the effect of shear stress and uses two-dimensional triangular elements to take the thickness variation into account. Friction, which is of primary significance compared with the secondary effect of bending and thickness, is assumed to be perfect, meaning that once the element touches the punch head, it does not slide over the punch but sticks to it.

2. Computational procedures

In applying the finite-element method to punch stretching, a thought should be given to the implementation of boundary conditions. The boundary conditions in punch stretching are stated not only by prescribing tractions and incremental displacements but sometimes by their ratios. In this report, the problem is similar to the ball indentation problem (Lee et al. [45]).

The radial and vertical positions of the material elements in the contact region are not independent but they are related to each other through a mathematical expression for the geometrical requirement that they must be actually on the surface of the punch head. The expression is

$$(r_0 + v)^2 + (c + z_0 + w)^2 = r_p^2, \tag{46}$$

where r_0, z_0 are radial and vertical positions of the element at the present undeformed configuration; v, w are the increments of horizontal and vertical displacements, and c is a parameter related to the punch height h by the expression

$$c = r_p - h.$$

See Fig. 11. Recall that the finite-element formulation in Chapter IV has already been linearized and what it really solves for are the perturbation terms. Therefore, we also linearize the boundary condition (46) to obtain

$$2(r_0 + v^*)\Delta v + 2(c + z_0 + w^*)\Delta w = r_p^2 - (r_0 + v^*)^2 - (c + z_0 + w^*)^2, \tag{47}$$

where starred (*) quantities are initial guesses, and Δv, Δw are perturbations. By rearranging (47), we have

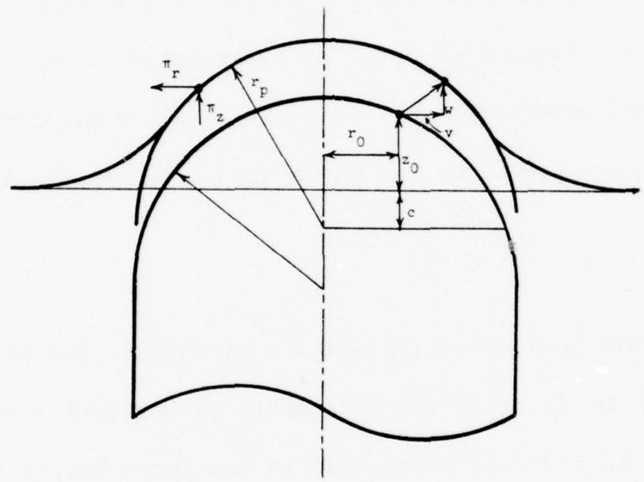


Figure 11. Geometrical Requirement for the Node on the Contact Region

$$\Delta v = \frac{1}{\alpha} \Delta w + \beta, \tag{48a}$$

where

$$\alpha = \frac{(r_0 + v^*)}{(c + z_0 + w^*)} = \frac{1}{\tan \theta} \tag{48b}$$

and

$$\beta = \frac{r_p^2 - (c + z_0 + w^*)^2 - (r_0 + v^*)^2}{2(r_0 + v^*)} . \tag{48c}$$

When the finite-element model is implemented, all the tractions are transformed into generalized nodal forces. Therefore, it is convenient to write the boundary condition in terms of the generalized nodal forces $\pi_{(r)}$ and $\pi_{(z)}$, the horizontal and vertical components, respectively. See Fig. 11.

Now

$$\begin{aligned} \pi_{(r)} &= N \cos \theta - S \sin \theta \\ \pi_{(z)} &= N \sin \theta + S \cos \theta \end{aligned} \tag{49}$$

where N and S are generalized forces normal and tangential to the punch head. We eliminate N through the relation

$$\cos^2\theta + \sin^2\theta = 1$$

and obtain

$$\pi_z \cos \theta = \pi_r \sin \theta + k, \tag{50}$$

where k is the frictional force at nodes. However, from geometry we know that the following holds:

$$\cos\theta = \frac{r_0 + v^*}{r_p}, \tag{51a}$$

$$\sin\theta = \frac{z_0 + w^* + c}{r_p}. \tag{51b}$$

So, (50) may be written as

$$\pi_{(z)} + \frac{\pi_{(r)}}{\alpha} = \frac{kr_p}{(r_0 + v^*)}. \tag{52}$$

If the die has a round profile of the radius r_D, then the requirement for a material element to lie geometrically on the profile is similar to the requirement to be satisfied on the punch head. Therefore, we have (similar to Eq. (46)),

$$(a - r_0 - v)^2 + (r_D - z_0 - w)^2 = r_D^2, \tag{53}$$

where a is the radius of the sheet. Linearization of Eq. (53) gives

$$2(a - r_0 - v^*)\Delta v + 2(r_D - z_0 - w^*)\Delta w = -r_D^2 + (a - r_0 - v^*)^2 + (r_D - z_0 - w^*) \tag{54}$$

or, rewriting,

$$\Delta v = \frac{\Delta w}{\gamma} + \Omega, \tag{55a}$$

where

$$\gamma = -\frac{(a - r_0 - v^*)}{(r_D - z_0 - w^*)} \tag{55b}$$

and

$$\Omega = \frac{(a - r_0 - v^*)^2 + (r_D - z_0 - w^*)^2 - r_D^2}{2(a - r_0 - v^*)}. \tag{55c}$$

The tractional boundary condition over the die profile can be written similarly as

$$\pi_{(z)} + \frac{\pi_{(r)}}{\gamma} = - \frac{k\gamma_D}{(a - r_0 - v^*)} \ . \tag{56}$$

For the portion of the sheet which is not in contact with the punch head nor with the die profile, the displacement increment in the radial direction and the displacement increment in the axial direction are not bound to each other, as is the case for the contact region, but remains as independent variables. Tractions are, however, given the value of zero.

With the advancement of the punch head, the portion of the sheet in contact with the punch or die profile increases and, consequently, the boundary separating this "contact region" from the "unsupported region" changes. The presence of this moving boundary is always a source of complications in the numerical analysis of punch stretching because it requires a basically trial-and-error approach. The treatment of the moving boundary used in the present analysis for punch stretching without round die corners is explained in detail as follows:

First, assume the position of the boundary in future configurations. In the FEM this means assuming which nodes will be in contact with the punch head in the future configuration. Then, obtain a converged solution based upon this assumption and check to see if it is true. Since the position of the boundary is already known in the current configuration, in practice we assume and check how much this boundary advances.

(1) Check whether the boundary is assumed to advance too fast.

Compute the normal component of the generalized nodal force to the punch head for the nodes in contact with the punch head. If every generalized normal force is directed outward from the punch head, then all the nodes which are assumed to be in contact with the punch head actually do so. On the other hand, the generalized normal force in the direction toward the punch head for any node means that external force other than the one exerted by the punch is necessary for this particular node to conform with the punch geometry. Since there is physically no source of applied force other than the punch, the assumption that this particular node is in contact with the punch head is not correct and the position of the boundary should be re-assumed to exclude this node from the contact region.

(2) Check whether the boundary is assumed to advance too slowly. Compute the distance between the nodes in the unsupported region and the center of the hemisphere of the punch head. If this distance is shorter than the radius of the punch head for any node, it means that this particular node is inside the punch head. Since this is physically impossible, the assumption that this particular node is not in contact with the punch head is not correct and the position of the boundary should be re-assumed to include this particular node.

Although this basically trial-and-error approach seems to be very time consuming, in actual computation we can predict the movement of the boundary fairly accurately based upon the distance between the free nodes and the punch surface. Furthermore, since we already know the position of the boundary in the current configuration, it is enough to check the boundary assumption for only a few nodes neighboring the previous position of the boundary, not for whole nodes.

The procedure described above for the contact region on a punch head is also applicable for the contact region at the die corner. Handling two moving boundaries simultaneously really does not invoke any new theoretical difficulties but only takes more computation time and may be impractical for inefficient numerical methods.

In order to implement Coulomb friction between sheet and punch or die, we first prescribe a tangential friction force S and obtain a converged solution and then compute generalized nodal forces. From Eqs. (49) we then are able to compute the normal component N and the friction coefficient $\mu = \frac{S}{N}$ corresponding to the initially prescribed value of S. If the computed friction coefficient is not what is intended, then we modify the S value and repeat the process. It should be noted here that the correction of frictional force S needs the necessary modification only in the F matrix (Eq. (44)), while the stiffness matrix P, which is the most time-consuming part, remains the same.

The deformation step is controlled by the punch head increment, which is designed in the present codes to yield the maximum increment of effective strain roughly equal to a preset value. In the present work the optimum size is shown to be a 0.04 increment of effective strain. The solution generally converged after $10 \sim 15$ iterations for a single step within the fractional norm of 10^{-6}. The actual program is shown in Appendix C.

3. Results and discussion

The present rigid-plastic FEM is compared with the finite-difference methods by Wang [43] and Woo [41], and also with the experiment by Kaftanoglu and Alexander [39].

(1) Comparison with the finite-difference solution by Wang

The parameters used in Wang's example are as follows:

Material: copper

Stress-strain characteristics: $\sigma = 30.5\varepsilon^{0.326}$ ton/in.^{2}

$= 4.6361\varepsilon^{0.326} \times 10^8 N/m^2$

Anisotropy: $R = 1.0$

Friction: $\mu = 0.04$

Thickness: $t = 0.035$ in. $= 8.89 \times 10^{-2}$ m

Punch radius: $r_p = 1.0$ in. $= 2.54 \times 10^{-2}$ m

Radius of sheet: $a = 1.15$ in. $= 2.921 \times 10^{-2}$ m

Initial radius is sometimes denoted by r_0.

The two methods are in excellent agreement in predicting the punch head for a given punch travel. See Fig. 12; the solid line represents Wang's solution and the points represent the rigid-plastic FEM. Fig. 13 shows the thickness strain distribution. Again, a good agreement between the two solutions is apparent.

The second example has the following parameters:

Stress-strain characteristics: $\sigma = k\varepsilon^{0.2}$

Anisotropy: $R = 1.0$

Friction: $\mu = 0.2$

Punch radius: $r_p = 1.0$

Radius of sheet: $r_0 = 1.0$

In Wang's work all the results are reported in the dimensionless number. Figs. 14 and 15 show the circumferential strain distribution and thickness strain distribution, respectively. The solid line represents Wang and points

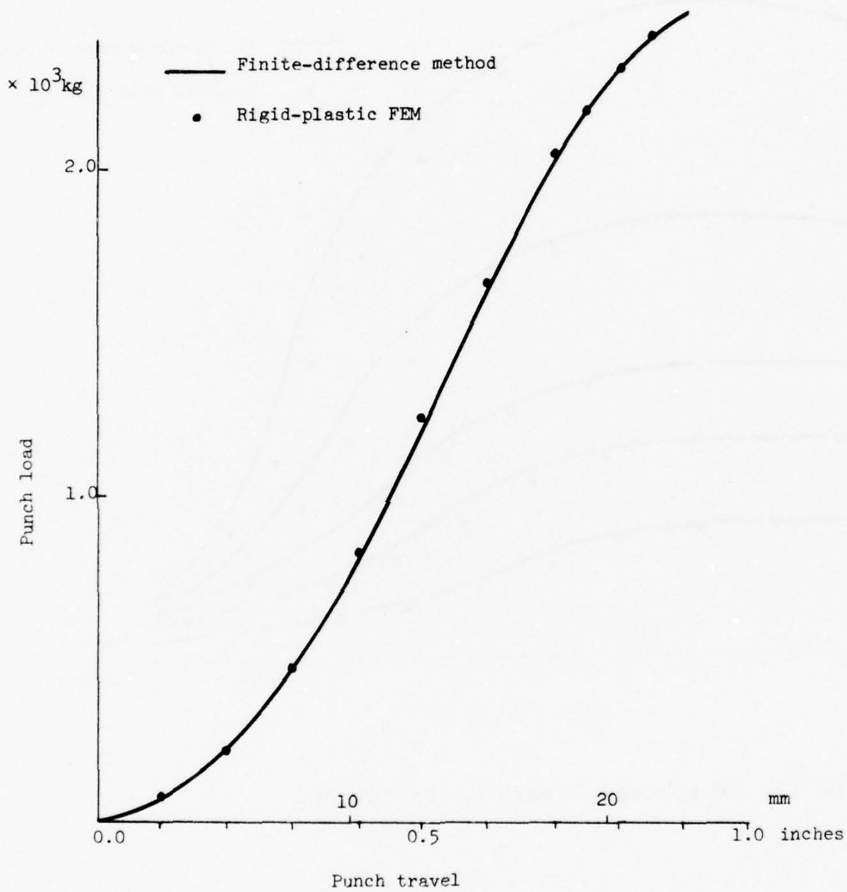


Figure 12. Punch Head vs. Punch Travel

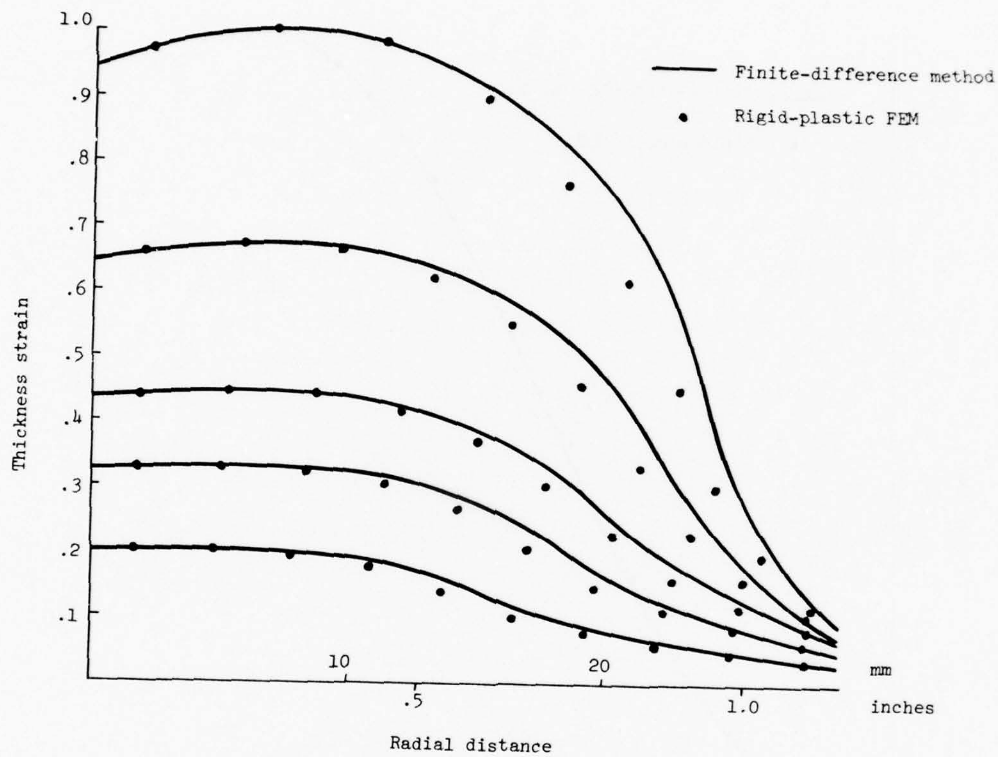


Figure 13. Thickness Strain Distribution

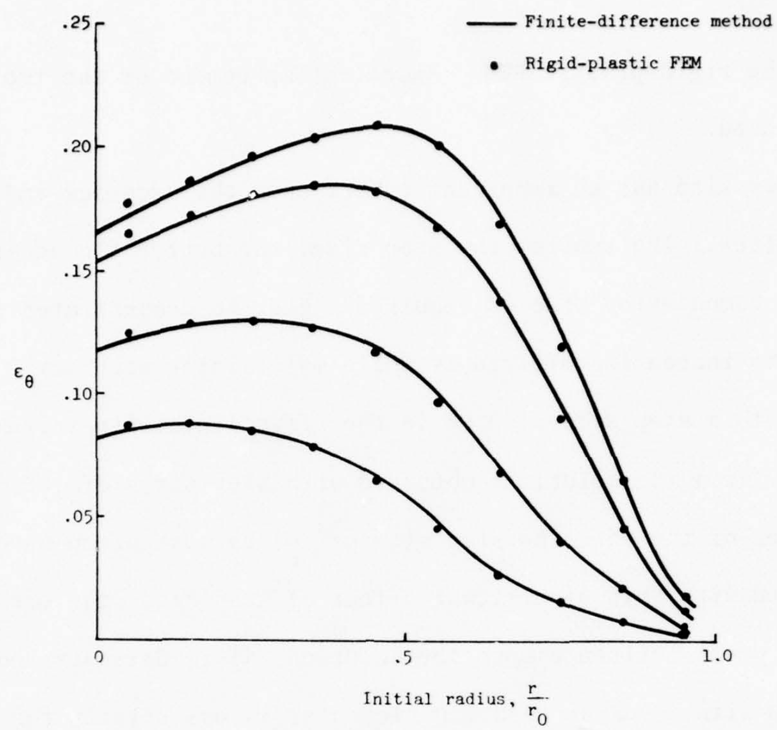


Figure 14. Circumferential Strain Distribution

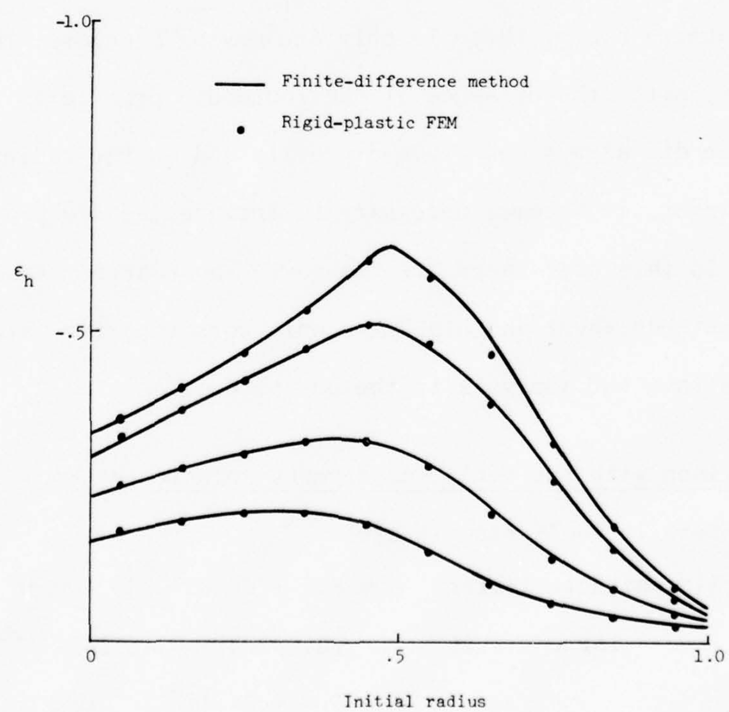


Figure 15. Thickness Strain Distribution

represent the rigid-plastic FEM. Excellent agreement of the two solutions is demonstrated.

The step size has an important effect upon the accuracy and efficiency of the solution. The smaller the step size, the better the accuracy, although more computation time is required. Fig. 16 demonstrates that there is a limit to increasing efficiency while maintaining accuracy. For example, solutions with a step size of 0.08 in the effective strain increment deviates considerably from the solutions obtained with step sizes of .02 or .04. In the remainder of the work the step size of .04 is most often used.

Compared with this significant effect of step size, the mesh size does not exert a great influence upon the solution, as is demonstrated in Fig. 17. The solution with a coarse mesh (10 elements) is essentially the same as the one with a finer mesh (40 elements), even though the latter will be helpful in pinpointing the exact location of peak strain.

In the examples above, there is only one moving boundary, that between punch and sheet, since the presence of the round die profile is neglected. In practice, the die always has a round profile and as the radius of the profile gets larger, it becomes necessary to include the die profile in the analysis. In this case there are two moving boundaries, the second being the one between sheet and die. The only work reported which includes the die profile into the analysis is the one by Woo.

(2) Comparison with the finite-difference solution by Woo

The parameters in Woo's example are:

Stress-strain characteristics: $\sigma = 5.4 + 27.8\varepsilon^{0.504}$ ton/in.^{2}

for $\varepsilon < 0.36$: $= (0.08208 + 0.422569\varepsilon^{0.504}) \times 10^9 N/m^2$

$= 5.4 + 24.4\varepsilon^{0.375}$ ton/in.^{2}

for $\varepsilon > 0.36$: $= (0.08208 + 0.37089^{\ 0.375}) \times 10^9 N/m^2$

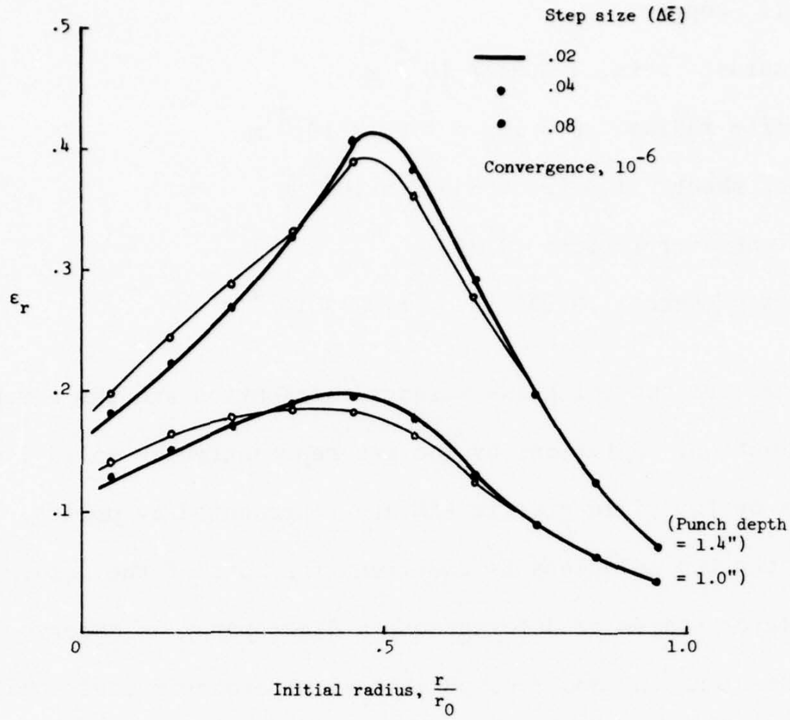


Figure 16. Effect of Step Size

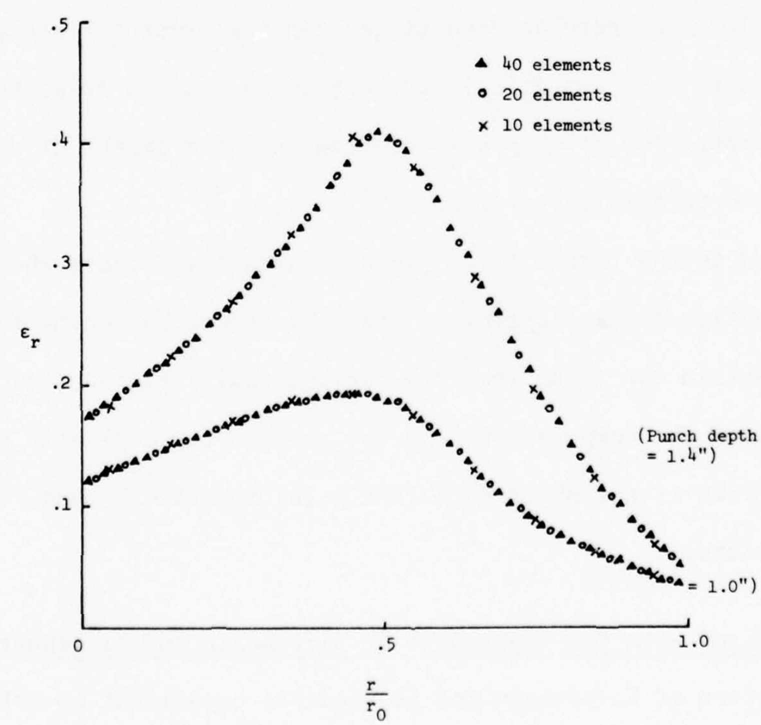


Figure 17. Effect of Mesh Size

Material: copper

Punch radius: 1 in. = 2.54×10^{-2} m

Die profile radius: 0.3 in. = 7.62×10^{-3} m

Radius of sheet: 1.3 in. = 3.302×10^{-2} m

Coefficient of friction: 0.04

Thickness of sheet: 0.035 in. = 8.89×10^{-4} m

Figs. 18 and 19 are the thickness strain distribution and the circumferential strain distribution. Solutions by Woo are represented by solid lines and the solutions by the rigid-plastic FEM are represented by points. Agreement between the two solutions is excellent for most of the deformation. However, at later stages of deformation, a discrepancy is observed around the edges. Re-examining Woo's computational procedure reveals that in order to avoid the difficulty of satisfying boundary conditions exactly along the fixed edge ($\varepsilon_\theta = 0$), he allowed a small increment of circumferential strain along the edge at each stage. In the present rigid-plastic FEM such difficulty does not exist, so there is no need to relax the boundary condition. The discrepancy observed at later stages of deformation may be attributed to this difference.

With regard to the instability, Woo stated that it occurs when the resultant tangential stress determined from the strain hardening characteristics cannot obtain the value required for the equilibrium and at that instant he stopped the computation. In the present rigid-plastic analysis such an instability is not observed at the point reported by Woo, and the computation continues.

(3) Comparison with the experiment by Kaftanoglu and Alexander

The parameters of Kaftanoglu and Alexander's experiment on soft copper are:

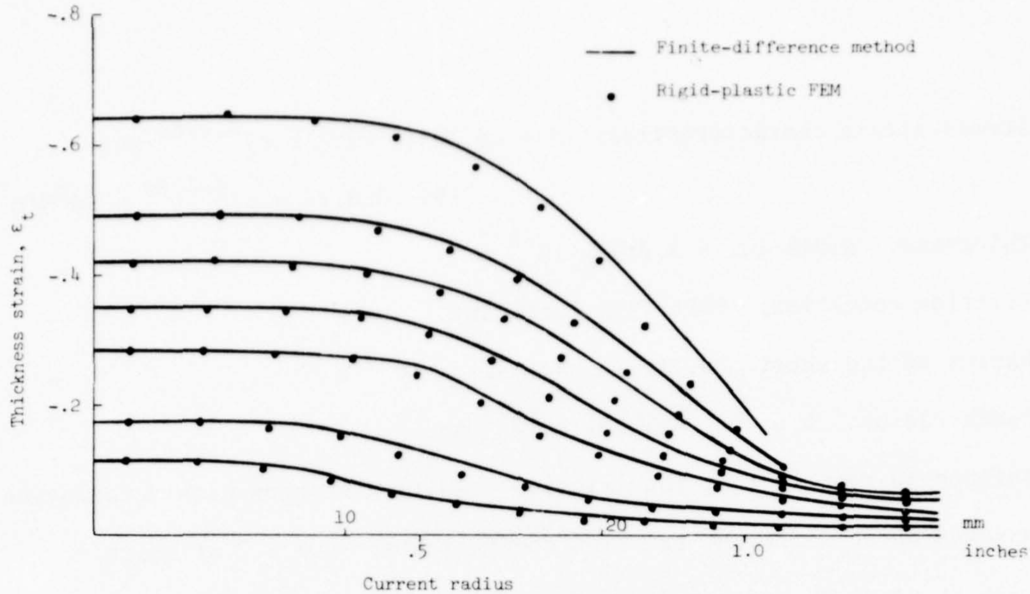


Figure 18. Distribution of Thickness Strain When Die Profile Is Considered

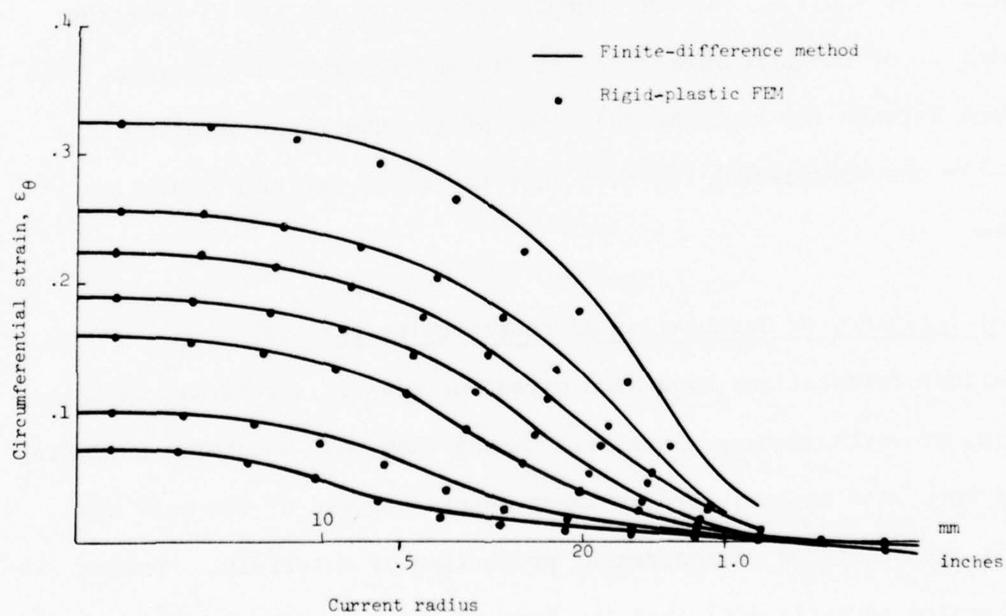


Figure 19. Distribution of Circumferential Strain When Die Profile Is Considered

Stress-strain characteristics: $\sigma = 68{,}394(0.0122 + \varepsilon)^{0.3789}$ psi

$= 4.7156\ (0.0122 + \varepsilon)^{0.3789} \times 10^{8}\,N/m^{2}$

Thickness: 0.048 in. = 1.219×10^{-3} m

Friction condition: PTFE film lubricant

Radius of the sheet: 0.717 in. = 1.821×10^{-3} m

Punch radius: 0.65 in. = 1.651×10^{-3} m

Kaftanoglu reports that the friction condition changes with deformation and measures three different friction coefficients: $\mu = 0.2$ at stage 1, $\mu = 0.135$ at stage 2, and $\mu = 0.07$ at stage 3. To include the changing friction coefficient into the analysis, we need more information on the friction history, which is difficult to obtain experimentally. Therefore, as a representative value, we use the mean of three values of the friction coefficient, $\mu = 0.135$, for our computation. Figs. 20 and 21 show the distribution of the circumferential strain and the thickness strain. The agreement between the experimental data and the numerical solution is a reasonable one considering the fact that the exact friction condition is not known.

(4) Influence of formulation of constitutive relation

Various formulations have been given for plastic stress-strain relationships of workhardening materials. Among them, the parabolic hardening law has been used extensively for sheet metals because of the ease with which it characterizes workhardening properties of materials. However, it was suggested recently [46] that the Voce equation [47] is a better representation of materials behavior when solving plasticity problems involving workhardening rate. The forming limit curves were compared using the

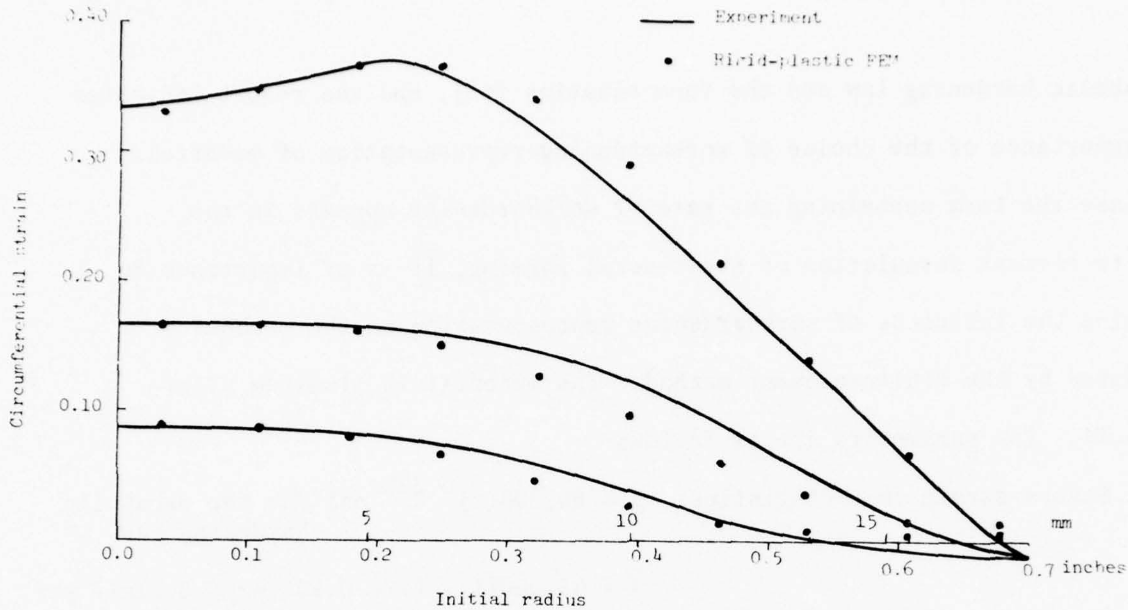


Figure 20. Comparison of the Numerical Solution with the Experimental Data for Circumferential Strain Distribution

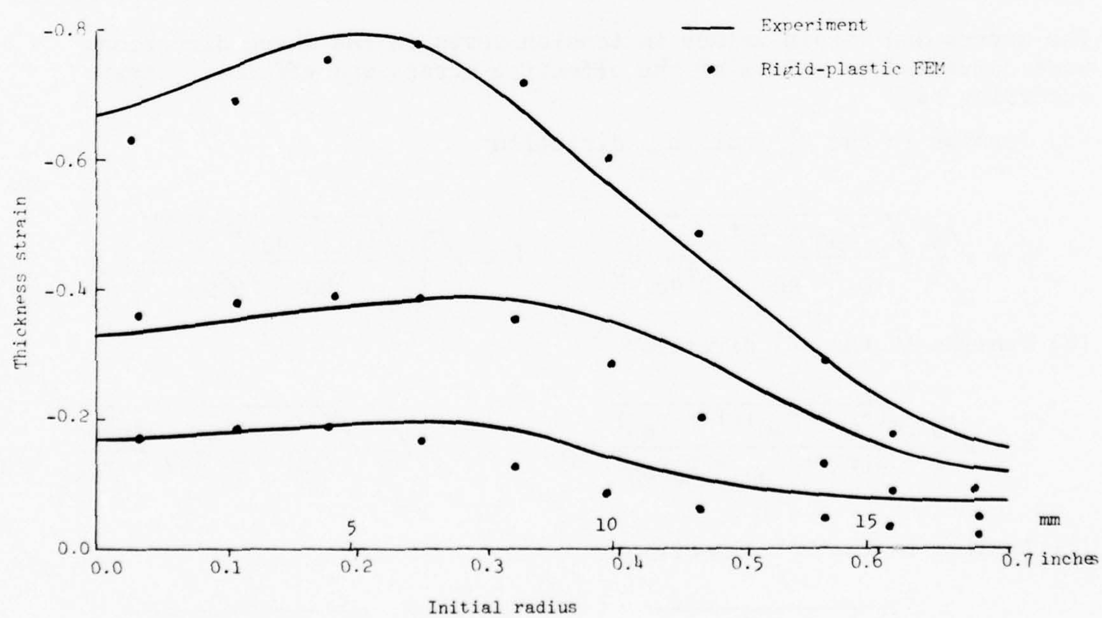


Figure 21. Comparison with the Experiment for Thickness Strain Distribution

parabolic hardening law and the Voce equation [48], and the result indicates an importance of the choice of workhardening representation of materials. Because the term containing the rate of workhardening appears in the finite-element formulation of sheet-metal forming, it is of importance to examine the influence of workhardening representation on the mechanics computed by the finite-element method. The material is aluminum alloy 2036-T4. The parameters are as follows:

Stress-strain characteristics: $\sigma = 86{,}000(\varepsilon)^{0.222}$ psi for the parabolic hardening law

$\sigma = 65{,}000\{1 - (1 - 0.508)\exp(-8.51\varepsilon)\}$ psi for the Voce equation

Fig. 22 shows the two stress-strain curves together with tension test data from the specimens cut in the three directions (0°, 45°, 90°).†

---

†(1) The stress and strain values in tension tests in the three directions were converted to values of the effective stress and effective strain according to

(a) Tension in the 0° (rolling) direction:

$$\bar{\sigma} = \sqrt{\frac{3}{2}} \sqrt{\frac{r_{90} + r_0 r_{90}}{r_0 + r_{90} + r_0 r_{90}}}\, \sigma_0, \qquad \bar{\varepsilon} = \sqrt{\frac{2}{3}} \sqrt{\frac{r_0 + r_{90} + r_0 r_{90}}{r_{90} + r_0 r_{90}}}\, \varepsilon_0;$$

(b) Tension in the 45° direction:

$$\bar{\sigma} = \sqrt{\frac{3}{2}} \sqrt{\frac{(r_0 + r_{90})(1 + r_{45})}{2(r_0 + r_{90} + r_0 r_{90})}}\, \sigma_{45}, \qquad \bar{\varepsilon} = \sqrt{\frac{2}{3}} \sqrt{\frac{2(r_0 + r_{90} + r_0 r_{90})}{(r_0 + r_{90})(1 + r_{45})}}\, \varepsilon_{45};$$

(c) Tension in the 90° direction:

$$\bar{\sigma} = \sqrt{\frac{3}{2}} \sqrt{\frac{r_0 + r_0 r_{90}}{r_0 + r_{90} + r_0 r_{90}}}\, \sigma_{90}, \qquad \bar{\varepsilon} = \sqrt{\frac{2}{3}} \sqrt{\frac{r_0 + r_{90} + r_0 r_{90}}{r_0 + r_0 r_{90}}}\, \varepsilon_{90};$$

(Footnote continued on next page)

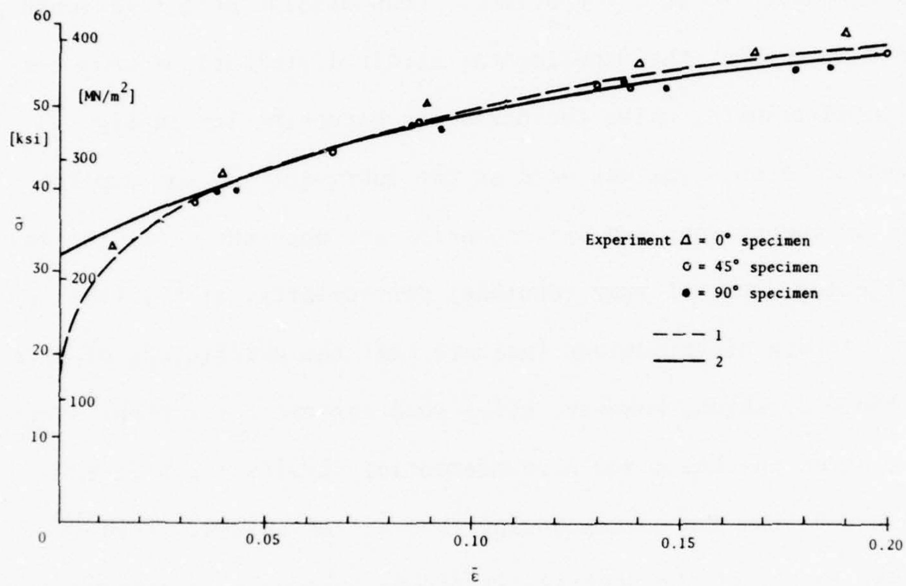


Figure 22. Stress-strain Curve for Al. 2036-T4

r-value: $r_0 = 0.66$, $r_{45} = 0.69$, $r_{90} = 0.70$, and $r_a = 0.685$

Radius of die opening: (0.80 in.)

Blank thickness:

Radius of punch head: 0.75 in. and 0.45 in.

Coefficient of friction: 0 and 0.2

Punch stretching was performed on a horizontal hydraulic press. Tests were interrupted for strain measurements (thickness and circumferential strains) from the grids photoprinted on the specimen. Load-displacement relationships were also recorded. First, the experimental strain distributions were compared with computed results, using the parabolic hardening law in Fig. 23. In the experiment Johnson's wax was used as the lubricant and was applied at each stage. In comparison, two discrepancies are apparent: (i) the coefficient of friction does not stay constant; particularly, at the last stage, the experimental strain distributions indicate that the coefficient of friction is less than 0.2, which, however, gives good agreement for other stages, and (ii) the measured thickness and circumferential strains for a given punch depth do not follow the corresponding theoretical curves. This is attributed to the fact that the accurate strain measurements is extremely difficult for critical comparison between theory and experiment. The load values summarized in Table 1 show an excellent agreement between the two.

---

where r_0, r_{45}, r_{90} are the r-values obtained from the tension of specimens cut in the 0°, 45°, and 90° directions, respectively.

(2) The effective stress and effective strain defined in the formulation of this report differ from the definition above by a factor such as

$$\bar{\sigma} = \sqrt{\frac{3}{2}\frac{1 + r_a}{2 + r_a}}\ , \qquad \bar{\varepsilon} = \sqrt{\frac{2}{3}\frac{2 + r_a}{1 + r_a}}\ ,$$

where r_a is the average r-value defined by $r_a = \dfrac{r_0 + 2r_{45} + r_{90}}{4}$.

Table 1

PUNCH LOAD AND DISPLACEMENT RELATIONS

Theoretical				Experimental			
Displacement mm	(in.)	Punch load N	(lb)	Punch load (lb)	N	Displacement	
Punch head radius = 19.05 mm (0.75 in.)							
4.06	(0.160)	6,330	(1,423)	(970)	4,315	2.79	(0.110)
6.10	(0.240)	10,889	(2,448)	(1,730)	7,695	4.83	(0.190)
7.54	(0.297)	14,483	(3,256)	(2,990)	13,300	7.11	(0.280)
9.80	(0.386)	22,059	(4,959)	(4,940)	21,974	10.08	(0.397)
12.45	(0.490)	30,301	(6,812)	(6,580)	29,269	12.45	(0.490)
Punch head radius = 11.43 mm (0.45 in.)							
4.06	(0.160)	5,124	(1,152)	(920)	4,092	2.72	(0.107)
6.10	(0.240)	8,131	(1,828)	(2,000)	8,896	6.30	(0.248)
8.53	(0.336)	12,237	(2,751)	(2,770)	12,322	8.18	(0.322)
9.58	(0.377)	13,963	(3,139)	(3,130)	13,923	9.68	(0.381)

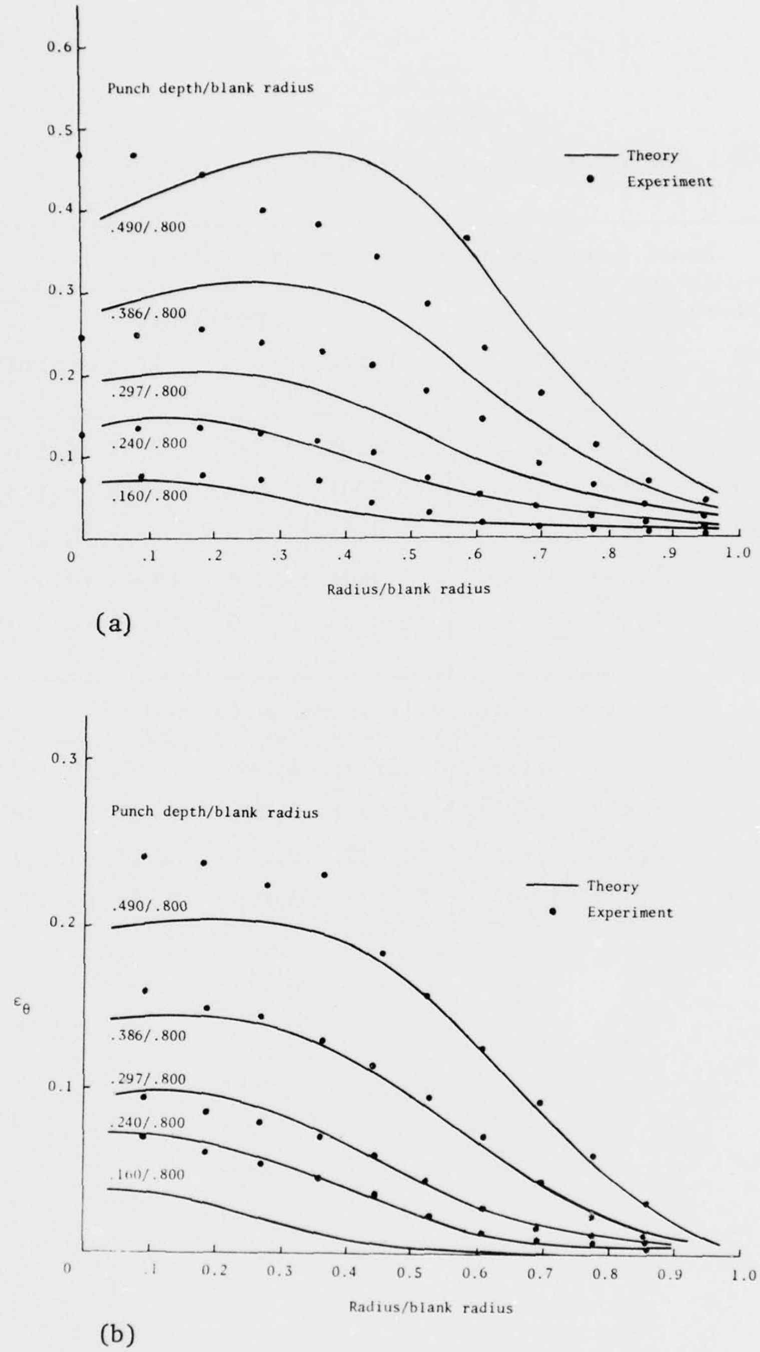
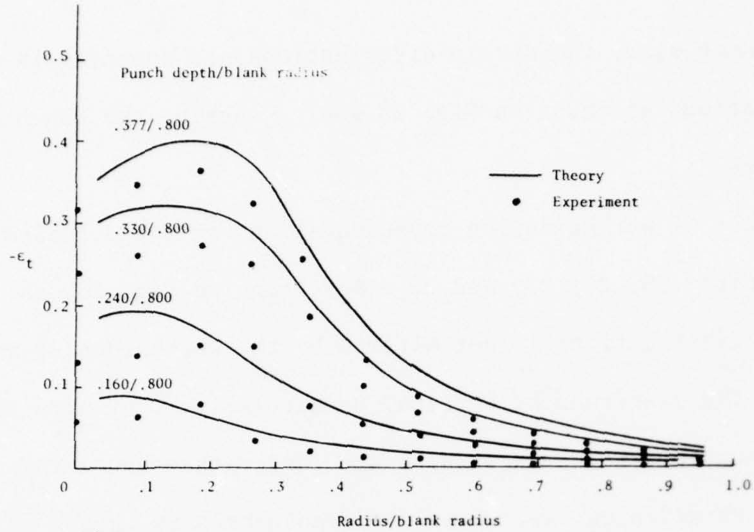
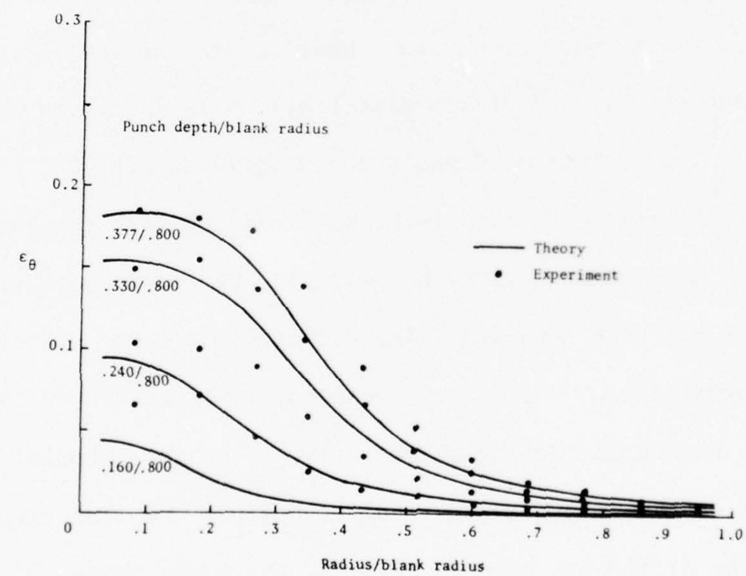


Figure 23. Experimental (Johnson's wax as lubricant) and Theoretical (μ = 0.2) Strain Distributions for Punch Size (r_p/r_0 = 0.75/0.80). (a) Thickness Strains; (b) Circumferential Strains



(a)



(b)

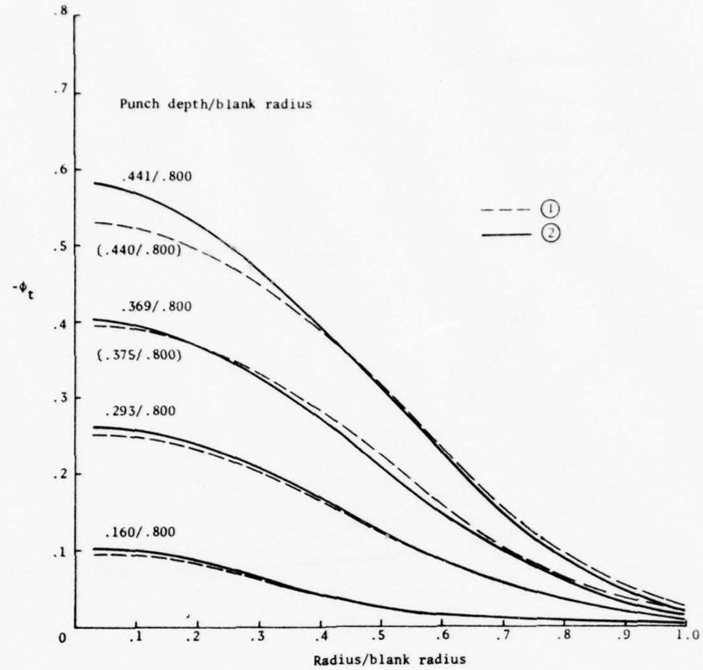
Figure 24. Experimental (Johnson's wax as lubricant) and Theoretical (μ = 0.2) Strain Distributions for Punch Size (r_p/r_0 = 0.45/0.80). (a) Thickness Strains; (b) Circumferential Strains

For a smaller punch size, the strain distributions are compared in Fig. 24. The same observations as those in Fig. 23 apply. Again, the punch load is in good agreement.

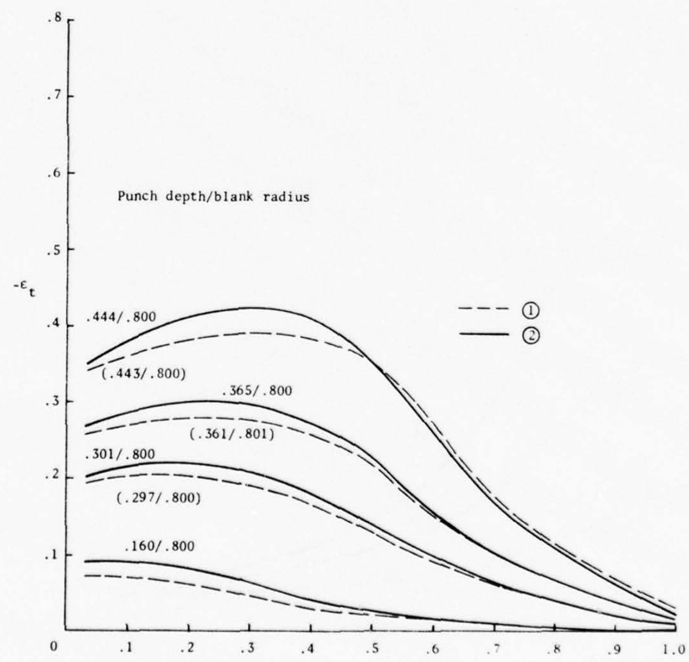
The influence of workhardening representations on the detailed mechanics is examined in Figs. 25, 26, 27, and 28. Referring to Fig. 25, the general trend of strain distributions is not altered by the workhardening representation. However, the magnitude of strains, particularly, peak strains, differ. With the Voce equation, the peak strains are larger than those computed by the parabolic workhardening law. This difference becomes larger as the punch penetrates.

It appears that the difference of the two is more significant for higher friction in the larger punch size. However, in the smaller punch size, the difference of the two strain distributions is about the same for the two coefficients of friction, 0 and 0.2, as shown in Fig. 26.

It is rather surprising to find in Figs. 27 and 28 that the punch load for the same punch displacement is higher with the parabolic workhardening law than that with the Voce equation. The difference becomes significant for large punch penetration. From these results, it is concluded that the representation of the workhardening characteristics of the material does have an influence on the computed strain distributions and load-displacement relationships. The difference becomes critical for large punch displacement in predicting both peak strains and the punch load. In order to determine which representation is preferable, however, more experiments with improved accuracy and control are needed.



(a)



(b)

Figure 25. Comparison of Theoretical Thickness Strain Distributions Using (1) the Parabolic Work-hardening Law and (2) the Voce Equation for Punch Size (r_p/r_0 = 0.75/0.80) with (a) μ = 0 and (b) μ = 0.2

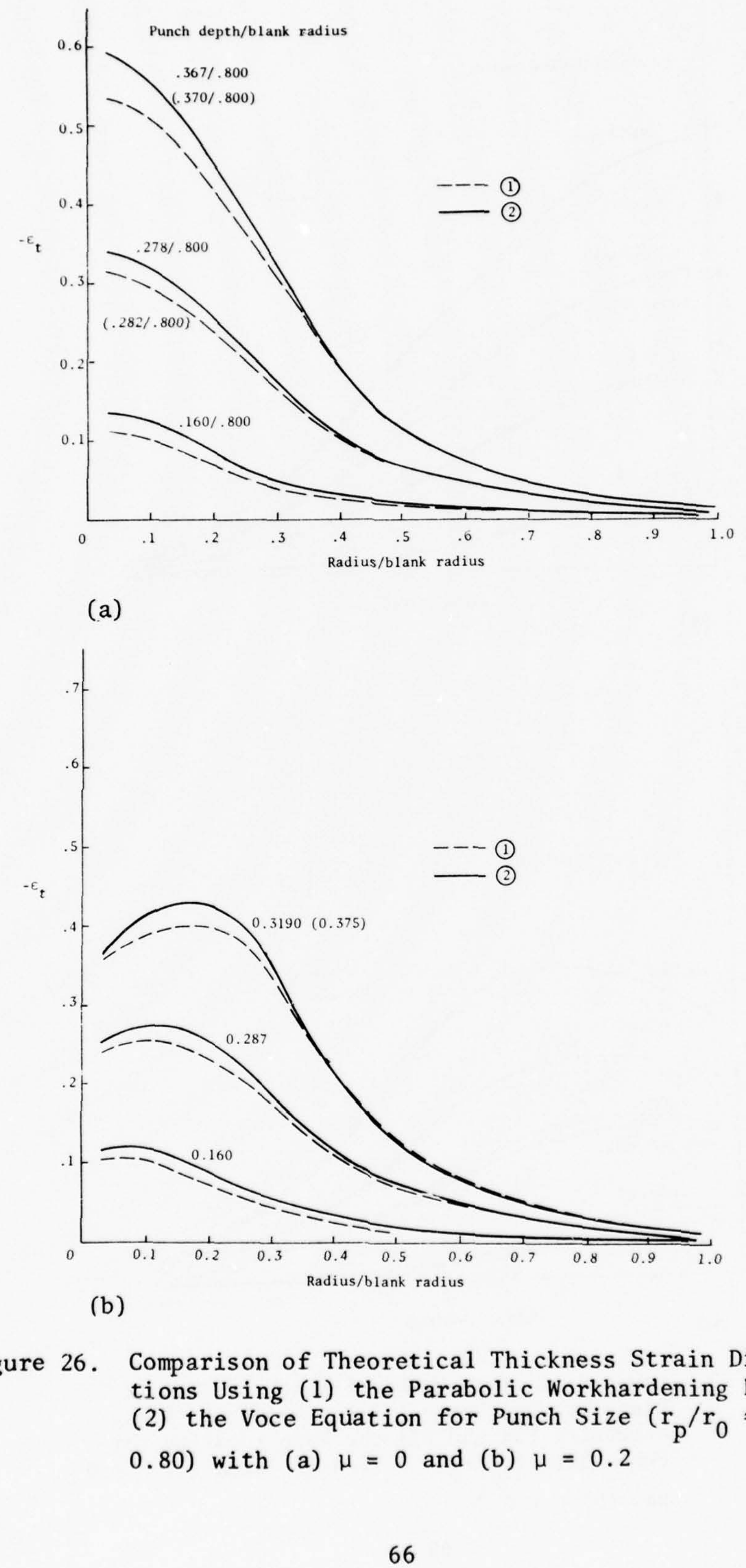


Figure 26. Comparison of Theoretical Thickness Strain Distributions Using (1) the Parabolic Workhardening Law and (2) the Voce Equation for Punch Size (r_p/r_0 = 0.45/0.80) with (a) μ = 0 and (b) μ = 0.2

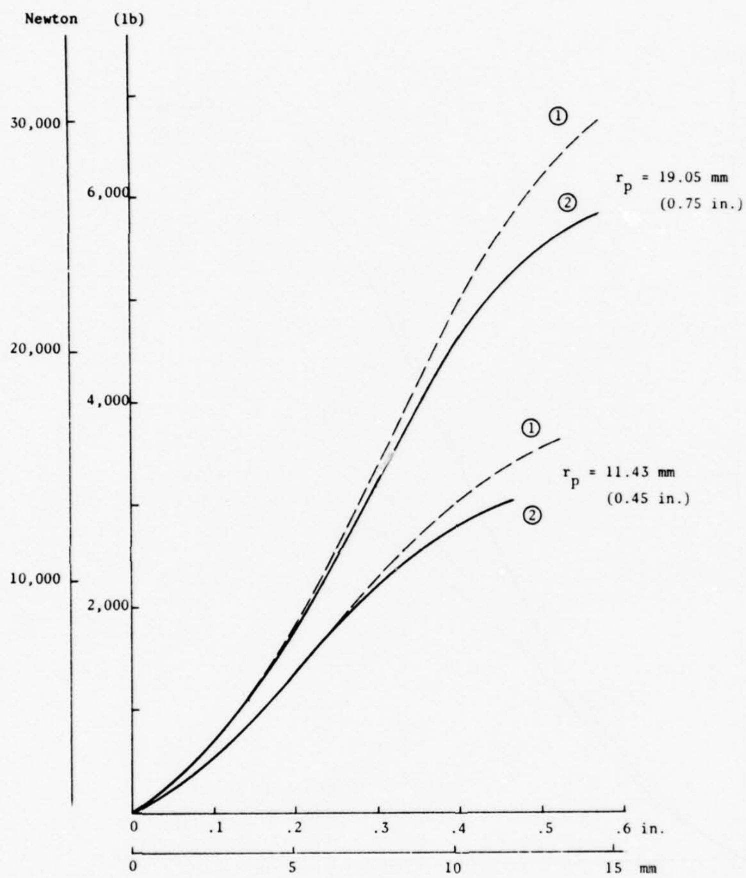


Figure 27. Comparison of Theoretical Load Displacement Curves Using (1) the Parabolic Workhardening Law and (2) the Voce Equation for $\mu = 0$

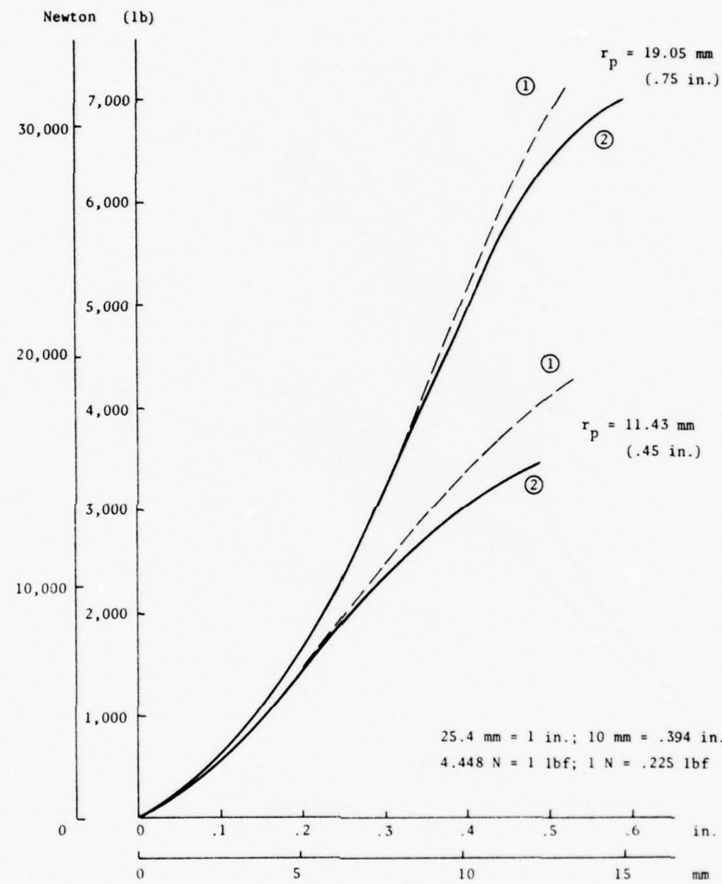


Figure 28. Comparison of Theoretical Load Displacement Curves Using (1) the Parabolic Workhardening Law and (2) the Voce Equation for $\mu = 0.2$

SECTION VI

DEEP DRAWING OF A SHEET WITH HEMISPHERICAL PUNCH

1. Introduction

In a deep drawing test a circular sheet of metal is placed between the blank holder and the die and then fully drawn into the shape of a cup. The formability is then measured by the maximum size of the blank which can be drawn without a failure, or, more often, by its ratio to the punch diameter. This ratio is called the limiting drawing ratio and this particular kind of test is called the Swift test.

Deep drawing is not only a useful method of material testing, but also one of the basic operations in sheet-metal stamping. In practice, various shapes are possible for the bottom of the punch; however, most past investigations are on deep drawing with a flat-bottomed punch [49]-[56].

Among the earlier works on deep drawing are those by Hill [13] and by Chung and Swift [52] using the incremental theory of plasticity. More refined analyses are the finite-difference solutions by Chiang and Kobayashi [57], b- Wang and Budiansky [51], and by Chakrabarty and Mellor [49]. Even though such a refinement improves the understanding of the deep drawing process, their works are not complete because they treat the deep drawing problem as an in-plane pure radial drawing and are concerned mostly with the deformation mechanics on the flange. However, it has been observed experimentally (Chung and Swift [52]) that the die profile and the punch profile significantly affect the punch load and the strain distributions and therefore a further refinement is necessary by considering these parameters in the analysis. Woo [53] performs such an analysis and then is able to show that the solution obtained by extrapolating the strain distribution over the flange to the die throat predicts more straining than the one obtained by taking the profiles into consideration.

Contrary to these numerous investigations on deep drawing with a flat-bottomed punch, very few works are reported on the deep drawing of a sheet with a hemispherical head punch (Fig. 29). Woo [58] analyzes this problem by breaking down the deep drawing process into two component processes of the pure radial drawing over the flange and the punch stretching over the hemispherical punch head. He first obtains solutions for pure radial drawing in the flange and then uses this solution at a point initially situated near the die lip as the boundary condition for the stretching problem, and thereby essentially matched the punch stretching component with the pure radial drawing component at a particular point in the die profile region.

Instead of this tedious process of boundary matching, it is desirable to have a numerically efficient and reliable method which can treat the problem in a unified manner. The FEM is such an alternative. The finite-element model developed for the deep drawing problem is the one by Wifi [44] with a limited treatment of friction. Also, Levy et al. [59] developed the elasto-plastic finite-element program for cupdrawing based on the deformation theory of plasticity.

2. Computational procedure

The entire sheet undergoing the deep drawing process can be divided into four regions: the contact region with the punch head, the unsupported region, the contact region with the die profile, and the flange over the die. Different kinds of boundary restrictions are imposed depending upon

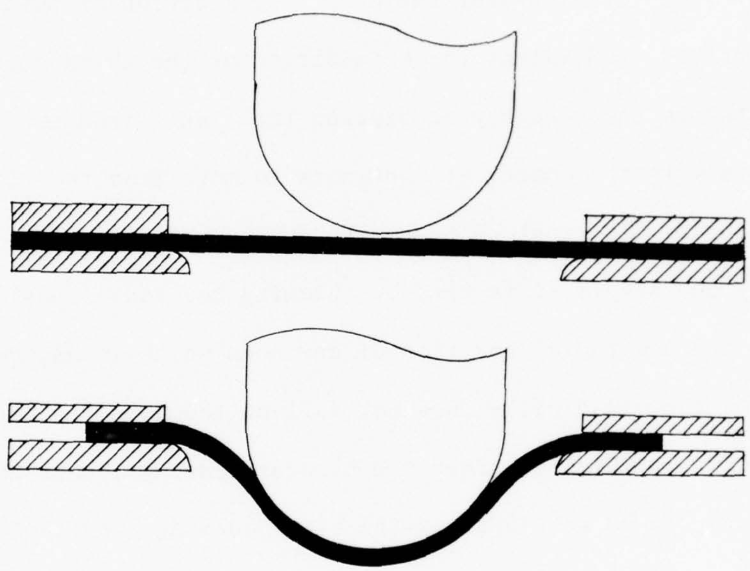


Figure 29. Schematic View of Deep Drawing of a Sheet with a Hemispherical Head Punch

the regions. For example, the flange is constrained to move only horizontally along the die face, while the contact region with the die profile or punch head should satisfy the kind of boundary conditions discussed in Section V.

The only difference in deep drawing with a hemispherical head punch from the punch stretching with a round die corner is the presence of the flange which is free to slide over the die. The addition of this moving flange is, in effect, equivalent to the addition of the third moving boundary, because, even though the boundary separating the flange from the die profile remains stationary in the space, it continues to move from the viewpoint of the deforming sheet. To treat this we make an assumption on this third moving boundary and see if it is true by checking the radial positions of the nodes. If the new radial position of any node which is assumed to lie on the flange or the die profile does not fall on the expected region after converged solution is obtained, then the boundary assumption is modified.

Another point to be mentioned is the blank holding condition of which there are two types: clearance holding and force holding. The idealization of the deformation state corresponding to the force blank holding is the plane stress state and the one corresponding to the clearance holding in the plane-strain state. The present rigid-plastic FEM is built to handle the plane stress state deformation and therefore a modification is necessary to handle the clearance blank holding. No reported work on deep drawing with a hemispherical head punch under clearance blank holding is available and therefore in the present work only the deep drawing with the force holding is analyzed. The blank holding force is implemented in the formulation as a tengential friction force acting on the last node located at the rim of the sheet. The distribution of the blank holding force over a finite area near the rim can be handled without difficulty in the present FEM, but this distributional effect turns out to be insignificant [53]. Therefore, tangential frictional force is confined to the last node at the rim of the sheet. The increment of deformation is controlled by the punch head movement. The program is in Appendix D.

3. Results and discussion

The only available work on the complete analysis of deep drawing with the hemispherical head punch is one by Woo [58]. Along with the numerical solution by the finite-difference method, he also conducted an experiment. The parameters are:

Material: soft copper

Stress-strain characteristics: $\sigma = 5.4 + 27.8\varepsilon^{0.504}$ ton/in.^{2}

for $\varepsilon < 0.36$: $= (0.08208 + 0.422569\varepsilon^{0.504}) \times 10^9 N/m^2$

$= 5.4 + 24.4\varepsilon^{0.375}$ ton/in.^{2}

for $\varepsilon > 0.36$: $= (0.08208 + 0.37089\varepsilon^{0.375}) \times 10^9 N/m^2$

Blank radius: 2.2 in. = 5.588×10^{-2} m

Radius of the die throat: 2.123 in. = 5.392×10^{-2} m

Radius of die profile: 0.5 in. = 1.27×10^{-2} m

Radius of punch head: 1 in. = 2.54×10^{-2} m

Blank holding force: 0.5 ton = 500 kg

The solution by the rigid-plastic FEM is in excellent agreement with the experiment for the flange part; however, over the punch head it predicts more straining than the experiment when the friction coefficient of 0.04 is assigned for the contact region over the punch head and over the die in the numerical analysis. When the friction coefficient is increased to a value of 0.1 over the punch head, while the same friction coefficient of 0.04 is used for the flange, the analysis predicts less straining over the punch head than the experiment. See Figs. 30, 31, 32, and 33. The deviation of the numerical solution from the experimental data gets larger as deformation progresses, which is reflected in the punch load vs. punch depth relationship in Fig. 34.

The lubricant used in the experiment is graphite in tallow and Woo suggested the friction coefficient to be 0.04. In the analysis the practical difficulty always lies in the assignment of a reasonable value of friction coefficient because friction coefficient under a real sheet-metal forming condition is hard to measure and it may even change during deformation.

Comparison of Woo's numerical solution with the experimental data does not yield any better agreement than the present rigid-plastic FEM. In comparing his numerical solution with the experiment Woo made the correction on the circumferential strain based upon the argument that the strain value obtained from the analysis is the value at the neutral surface of the sheet, while experimental data are obtained from the outside surface and therefore a compensation for the thickness difference is necessary. There could be a question about Woo's correction because the ratio of the punch radius or die profile radius to the sheet thickness is sufficiently large in his experiment that the membrane theory is justifiable. Besides, it seems a more consistent way to consider the problem in the three-dimensional stress state instead of the plane stress condition, which is the case used in Woo's analysis, if the variation of the strain across the thickness is to be taken into account.

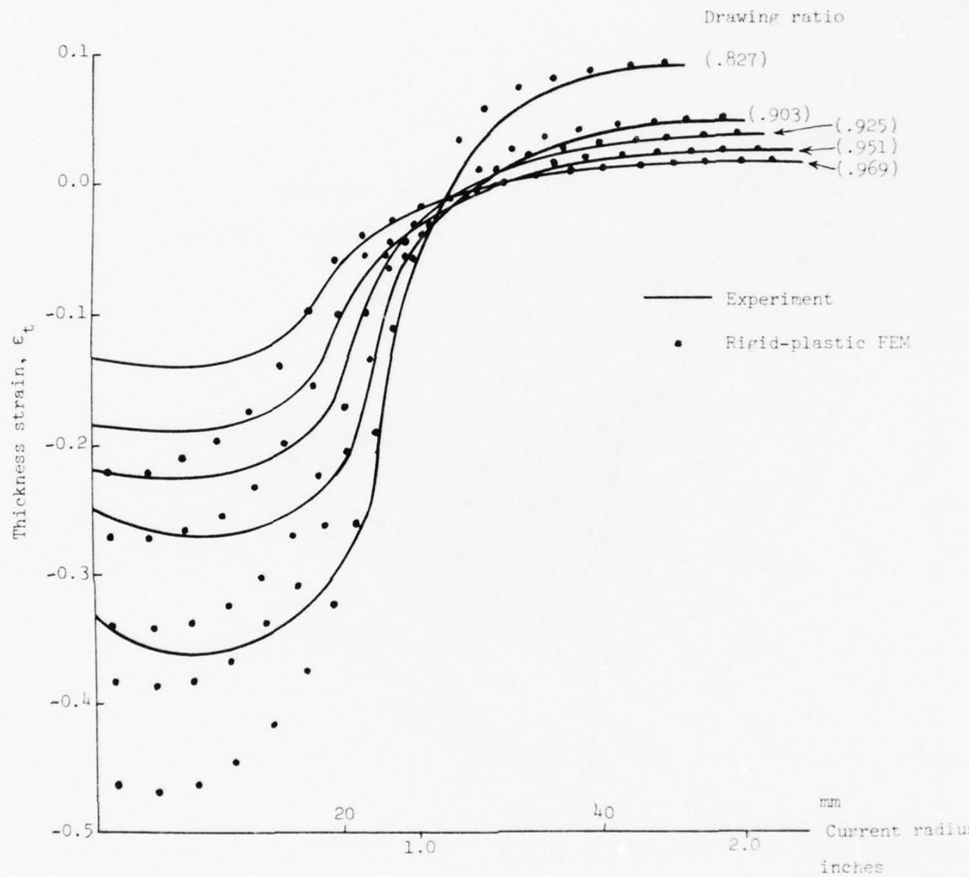


Figure 30. Distribution of Thickness Strain for $\mu_p = 0.04$, $\mu_d = 0.04$

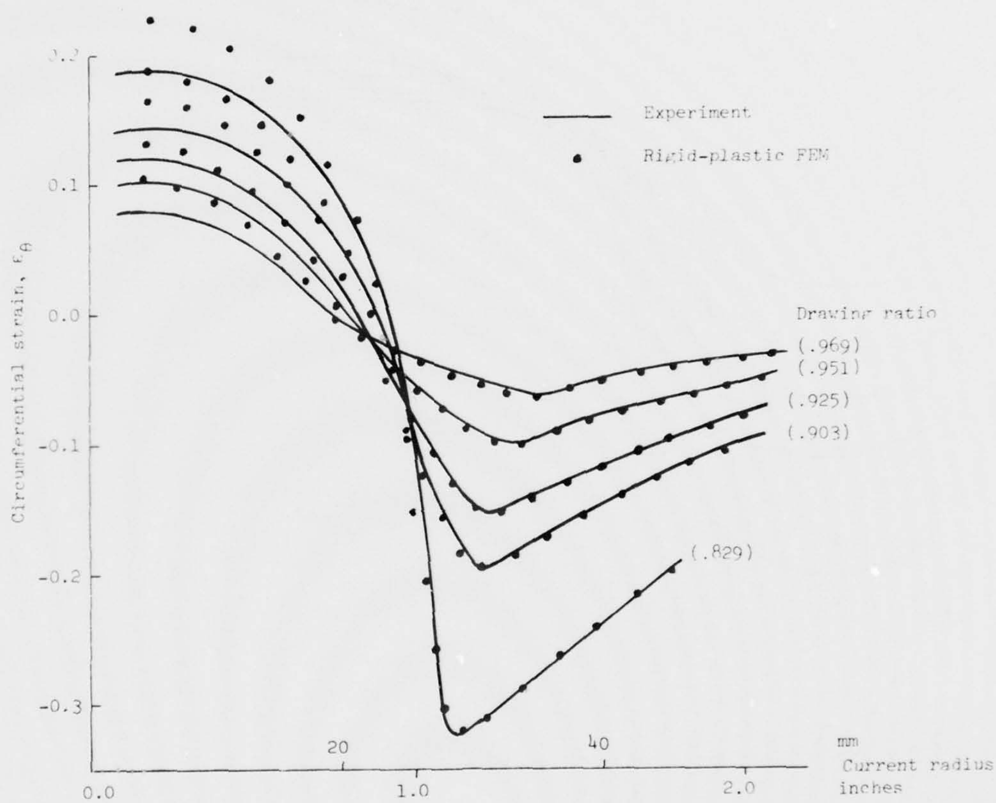


Figure 31. Distribution of Circumferential Strain for $\mu_p = 0.04$, $\mu_d = 0.04$

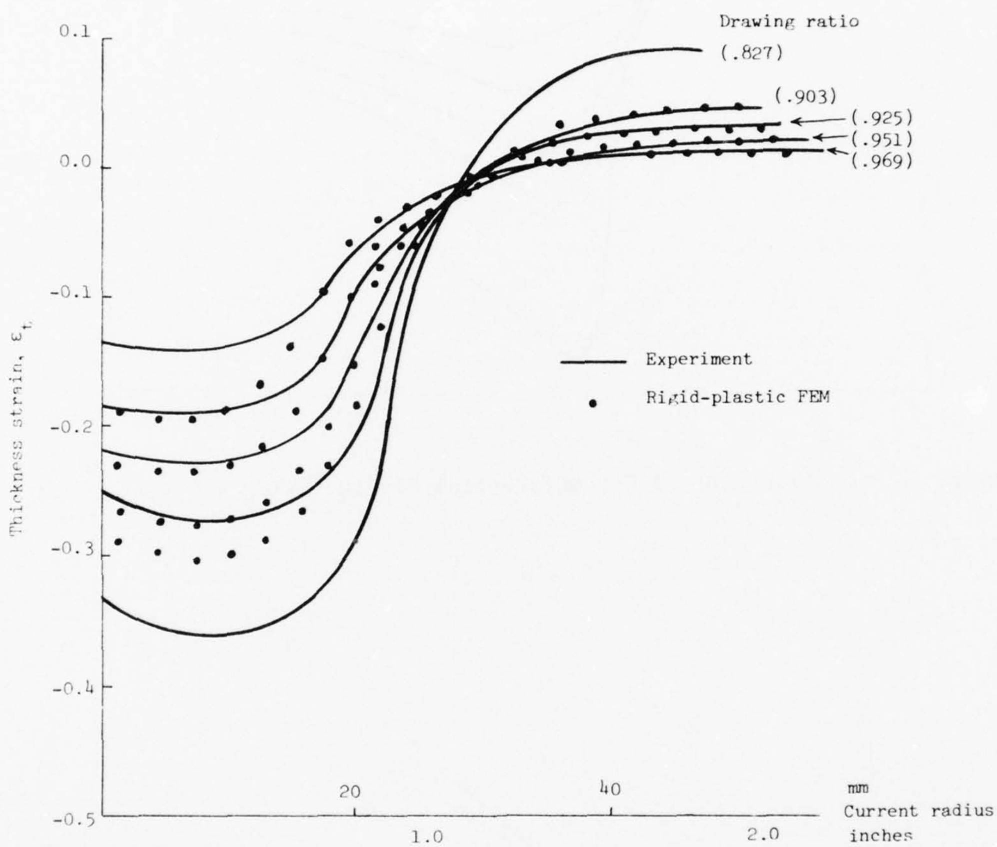


Figure 32. Distribution of Circumferential Strain for $\mu_p = 0.1$, $\mu_d = 0.04$

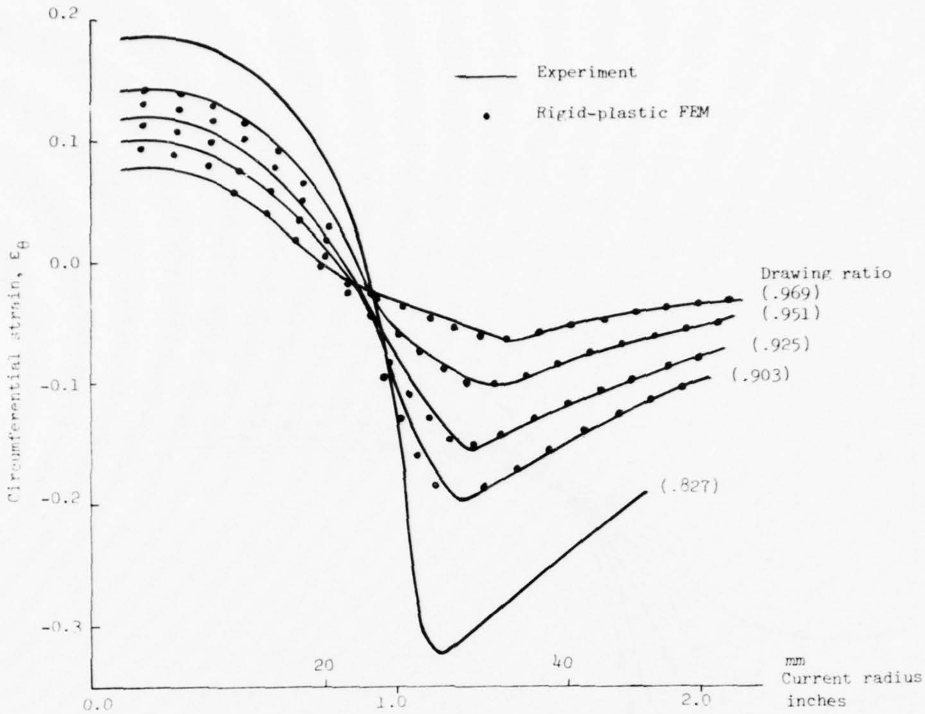


Figure 33. Distribution of Circumferential Strain for $\mu_p = 0.1$, $\mu_d = 0.04$

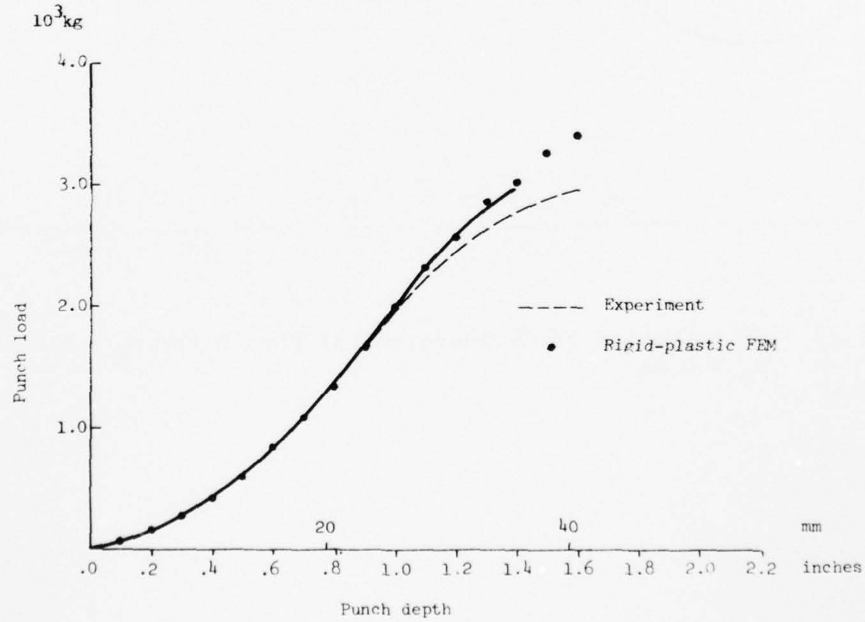


Figure 34. Punch Load vs. Punch Depth

It is necessary to have more numerical solutions and experimental data with a known friction state to assess the validity of the present rigid-plastic FEM for deep drawing problems. However, the present rigid-plastic FEM had dealt with other sheet-metal forming problems in a unified and consistent manner and therefore it seems reasonable to expect its validity for deep drawing problems when it is established for other problems.

SECTION VII

SUMMARY AND DISCUSSION

It has been made clear that classical variational formulations for the rigid-plastic solid are not appropriate for solving the sheet-metal forming problems. This is due to the nonuniqueness of the deformation mode under certain boundary conditions. This nonuniqueness, however, can be resolved by taking the workhardening rate into consideration. Such an introduction of the workhardening rate into the formulation, on the other hand, necessitates the consideration on the geometry change. The available classical formulation in which these two aspects are considered is not, however, applicable to the statically indeterminate problems, sheet-metal forming being one, because it is formulated in such a way that knowledge of stress distribution is necessary.

Within the framework of Eulerian descriptions and the hypothetical identity of the deformed configuration with the undeformed configuration, further improvement in the applicability of the variational formulations to the statically indeterminate problems is not possible. Therefore, an incremental deformation at a generic stage is considered by separating the deformed configuration from the undeformed configuration. The relevant equations are expressed with the undeformed configuration at each step as the reference frame and the variational formulation is established.

From this variational formulation a finite-element model is developed for the sheet-metal forming prlblems. In many sheet-metal forming processes the membrane theory is justifiable and therefore this idealization is introduced in building the model.

Three basic sheet-metal forming processes, i.e., the bulging of a sheet subject to the hydrostatic pressure, the stretching of a sheet with a hemispherical head punch, and deep drawing of a sheet with a hemispherical head punch are solved by the proposed method and its solutions are compared with the existing numerical solutions and the experimental data. The agreement is generally excellent and therefore the prime objective of the present investigation has been achieved.

In hydrostatic bulging the strain distributions and the pressure vs. polar height relationship predicted by the present rigid-plastic FEM are in excellent agreement with the available numerical solution by the elasto-plastic FEM and experimental data. The difficulty of satisfying the boundary condition along the fixed periphery experienced in the finite-difference method does not appear in the present rigid-plastic FEM.

In punch stretching, to make the problem more tractable, the presence of the die profile is neglected first so that there is only one moving boundary. This problem is successfully solved. Taking the die profile into consideration is equivalent to introducing another moving boundary, and while handling two moving boundaries simultaneously could be time consuming, the present rigid-plastic FEM again proves to be efficient and reliable. The strain distributions and the punch load vs. punch depth relationship predicted by the present rigid-plastic FEM are in excellent agreement with the numerical solutions by the finite-difference method and the experimental data.

We then investigate the influence of workhardening representation by comparing solutions, computed by both the parabolic workhardening law and the Voce equation methods. The two workhardening representations result in the difference of peak strains and load-displacement relationships, and the difference becomes increasingly significant as punch displacement increases. It is concluded, however, that the selection of a proper work-hardening representation requires more experiments with improved accuracy and control.

The present method is further extended to the deep drawing problem. The strain distribution predicted by the present rigid-plastic FEM is in excellent agreement with the experimental data over the flange of the sheet; however, over the punch head, agreement is not as good. By assigning two different values of the friction coefficient over the punch head, two strain distributions are obtained; one predicts more straining than the experimental data, and vice versa. Therefore, an improvement in the prediction seems possible by giving the friction coefficient a proper value which is between these two bounds; however, the validity of the present rigid-plastic FEM for deep drawing analysis remains inconclusive at this stage mostly because of the lack of comparable numerical solutions and experimental data. This is apparently due to the increased sophistication and accompanying computation time when three moving boundaries are treated simultaneously and to the practical difficulty of determining proper friction coefficients.

It is concluded that the present rigid-plastic FEM can treat the sheet-metal forming problems with efficiency and reasonable accuracy.

APPENDIX A

PROGRAM FOR THE INITIAL GUESS FOR HYDROSTATIC BULGING ANALYSIS

This program is to provide the initial guess and initial geometry for Appendix B. It is based upon the analysis by Hill [23].

(I) Data preparation

1. Read NUMNP (I5)

   NUMNP: Total number of nodal points to be generated

2. Read RADIUS, DIS1, DIS2 (3F 10.0)

   RADIUS: Radius of the sheet to be bulged

   DIS1: Polar height of the bulge in the initial geometry

   DIS2: Polar height of the bulge in the new configuration

```
      PROGRAM GRID(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,PUNCH)
C
C***********************************************************************
C     THIS PROGRAM IS TO GENERATE THE INITIAL GEOMETRY AND VELOCITY
C     FIELD FOR HYDROSTATIC BULGR PROBLEM, FOLLOWING HILL
C***********************************************************************
C
      COMMON A(2000)
C
C
C***********************************************************************
C     NUMNP=NUMBER OF NODAL POINTS TO BE GENERATED
C***********************************************************************
C
      READ(5,1001)NUMNP
C
C***********************************************************************
      N1=1
      N2=N1+NUMNP
      N3=N2+NUMNP
      N4=N3+NUMNP
      N5=N4+NUMNP
      N6=N5+NUMNP
      N7=N6+NUMNP
      N8=N7+NUMNP
      N9=N8+NUMNP
      N10=N9+NUMNP
      N11=N10+NUMNP
C
C
      CALL GUESS(A(N1),A(N2),A(N3),A(N4),A(N5),A(N6),A(N7),A(N8),A(N9),
     1A(N10),NUMNP)
C
 1001 FORMAT(I5)
      STOP
      END


      SUBROUTINE GUESS(RR,ZZ,CODE,SLOP,R,Z,UR,UZ,UUR,UUZ,NUMNP)
      DIMENSION RR(1),ZZ(1),CODE(1),SLOP(1),R(1),Z(1),UR(1),UZ(1),UUR(1)
     1,UUZ(1)
C
C
      READ(5,1001)RADIUS,DIS1,DIS2
      NUMEL=NUMNP-1
      K=0
      DIS=DIS1
      DR=RADIUS/FLOAT(NUMNP-1)
C
C
   50 K=K+1
      IF(K .EQ. 2)DIS=DIS2
      RR=RADIUS
      DO 100 I=1,NUMNP
      IF(K .EQ. 1)R(I)=RR
      IF(K .EQ. 1)Z(I)=0.
      IF(RR .EQ. 0.)GO TO 100
      R1=(RADIUS*RADIUS/RR-RR)/2.
      R2=(RADIUS*RADIUS/DIS+DIS)/2.
      DIS1=R2-DIS
      UR(I)=(R2*R2-R1*RR-R2*DIS1)*R1/(R1*R1+R2*R2)
      UZ(I)=(R2*RR+R2*R1-R1*DIS1)*R1/(R1*R1+R2*R2)
      CODE(I)=0.
      SLOP(I)=0.
C
  100 RR=RR-DR
      UZ(NUMNP)=DIS
      UR(NUMNP)=0.
      IF(K .EQ. 2)GO TO 301
C
      DO 300 I=1,NUMNP
      UUZ(I)=UZ(I)
  300 UUR(I)=UR(I)
C
```

```
  301 CONTINUE
C
      IF(K .LT. 2)GO TO 50
C
      DO 200 I=1,NUMNP
      UR(I)=UR(I)-UUR(I)
      UZ(I)=UZ(I)-UUZ(I)
  200 CONTINUE
C
      CODE(1)=3.0
      CODE(NUMNP)=1.0
      SLOP(NUMNP)=0.0
C
      DO 500 I=1,NUMNP
      RR(I)=R(I)+UUR(I)
      ZZ(I)=Z(I)+UUZ(I)
      WRITE(6,1000)I,RR(I),ZZ(I),UUR(I),UUZ(I),SLOP(I)
      WRITE(5,1000)I,R(I),Z(I),UR(I),UZ(I),SLOP(I)
      PUNCH 1011,I,CODE(I),RR(I),ZZ(I),UR(I),UZ(I),SLOP(I)
  500 CONTINUE
C
 1017 FORMAT(4F20.15)
 1011 FORMAT(I5,F5.3,5F10.7)
 1001 FORMAT(3F10.0)
 1000 FORMAT(I5,6F10.7)
      RETURN
      END
```

APPENDIX B

PROGRAM FOR THE ANALYSIS OF HYDROSTATIC BULGING

This program is for the analysis of hydrostatic bulging.

(I) Data preparation

1. Read HED (A 12)

   Output title

2. Read RVALUE, T, ACOEF (5F 10.0)

   RVALUE: Normal anisotropy parameter
   Set 1.0 for isotropic material

   T: Initial thickness of blank

   ACOEF: Accelerating coefficient
   To start with, set 1.0

3. Read ITER, NREAD, ITCONT, NFORM, NPUNCH, NPRINT, FLIMIT (6I5, F 10.0)

   The program control card

   ITER: Number of iterations to be executed

   NREAD: 1, if new data are to be supplied;
   0, otherwise

   ITCONT: 0, if computation starts at the very beginning and first/second steps are included in the steps to be computed;
   1, otherwise

   NFORM: Number of steps to be computed

   NPUNCH: 1, if solution is to be punched at the end of each step;
   0, otherwise

   FLIMIT: Value of (error norm)/(solution norm) required for convergence. To start with, set this .000001

4. Read NUMNP (6 I 5)

   NUMNP: Number of nodal points

5. Read YVALUE, PRESTN, EXPNT, PRESTS (4F 10.0)

   Material characteristics are specified.

   Stress = YVALUE* (Strain + PRESTN)**EXPNT + PRESTS

6. Read PRES, DPRES (4F 10.0)

   PRES: Current pressure value

   DPRES: Increment of pressure

7. Read N, CODE(N), R(N), Z(N), UR(N), UZ(N), SLOP(N), (I5, F5.0, 5F 10.0)

   Nodal information

   N: Node number. Node number 1 is at the rim of the blank and the last node is at the pole

   R(N): Radial position of the node

   Z(N): Axial position of the node

   UR(N): Increment of displacement in radial direction

   UZ(N): Increment of displacement in axial direction

   SLOP(N): Slope of the element
   Set this 0.0

   CODE(N): Type of boundary conditions:
   1.0, if magnitude of UR(N) is fixed;
   2.0, if magnitude of UZ(N) is fixed;
   3.0, if magnitudes of UR(N) and UZ(N) are fixed;
   0.0, if neither the magnitude of UR(N) nor UZ(N) are fixed

   In subroutine PRELIM the interpolation of data is built in.

8. If NREAD = 1, the input data is to be placed behind nodal information cards

```
      PROGRAM BULGE(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,PUNCH)
C
C***********************************************************************
C     THIS PROGRAM IS TO ANALYZE THE HYDROSTATIC BULGE)
C***********************************************************************
C
      COMMON/GENCON/NUMNP,NUMEL,HED(12),DLL,NEQ,NFORM,YIELD,TEST,ITER,
     1NREAD,NPUNCH,NPRINT,RVALUE,T,MBAND,PNFAD,RADIUS,PRES,DPRES
      COMMON/MATERL/YVALUE,PRESTN,EXPNT,PRESTS
      COMMON/ISOTPY/RVAL1
C
C
C***********************************************************************
C     PROGRAM        IS FOR CONTROLLING THE DIMENSION OF THE COMPLETE
C     PROGRAM.  ITS PURPOSE IS TO PREVENT ASSIGNING A LARGER THAN
C     NECESSARY DIMENSION FOR ANY ARRAY THROUGH THE USE OF THE
C     FOLLOWING STATEMENT
C***********************************************************************
C
      COMMON A(5000)
C
      NFIELD=5000
C
C***********************************************************************
C     NFIELD IS THE DIMENSION OF ARRAY A.  ITS VALUE CAN BE DETERMINED
C     PRECISELY BY RUNNING THE PROGRAM ONCE.
C***********************************************************************
C
C
      TEST=1.
C
      READ(5,1000) HED
      READ(5,1004) RVALUE,T,ACOEF
      READ(5,1003) ITER,NREAD,ITCONT,NFORM,NPUNCH,NPRINT,FLIMIT
      READ(5,1003) NUMNP
      READ(5,1004) YVALUE,PRESTN,EXPNT,PRESTS
      READ(5,1004)PRES,DPRES
C***********************************************************************
C     HED=OUTPUT TITLE
C     RVALUE=VALUE OF THE ANISOTROPY PARAMETER
C     ACOEF=ACCELERATING OR DECELERATING COEFFICIENT OF CONVERGENCE
C     NREAD=0, IF TO BYPASS THE READING STATEMENT IN SUBROUTINE PLAST
C     ITCONT=0, IF COMPUTATION STARTS AT THE VERY BEGGINNING AND FIRST/
C                  SECOND STEPS ARE INCLUDED IN THE STEPS TO BE COMPUTED
C            =1, OTHERWISE
C             THIS INDEX IS RELATED TO THE DETERMINATION OF STEP SIZE
C     NFORM=NUMBER OF STEPS ASSIGNED PER RUN
C     NPUNCH=1, IF DATA ARE TO BE PUNCHED
C            =0, OTHERWISE
C     FLIMIT=VALUE OF (ERROR NORM)/(SOLUTION NORM) REQUIRED
C     FOR CONVERGENCE
C     NPRINT=1, IF NODAL POINT DATA ARE TO BE PRINTED
C            =0, OTHERWISE
C     NUMNP=NUMBER OF NODAL POINTS
C     PRES= CURRENT PRESSURE
C     DPRES=INCREMENT OF THE PRESSURE
C
C     YVALUE, PRESTN, EXPNT, PRESTS ARE TO EXPRESS THE WORKHARDENING
C     CHARACTERISTICS OF THE BLANK
C     STRESS=YVALUE*(PRESTN+STRAIN)**EXPNT+PRESTS
C
C     NEQ=NUMBER OF EQUATIONS TO BE SOLVED
C     NUMEL=NUMBER OF ELEMENTS
C     MBAND=BAND WIDTH
C
C***********************************************************************
C
C
      NUMEL=NUMNP-1
      RVAL1=RVALUE
      MBAND=6
      NEQ=NUMNP*3
      NQ=NEQ
      NEL=NUMEL
C
      N1=1
      N2=N1+NUMNP
```

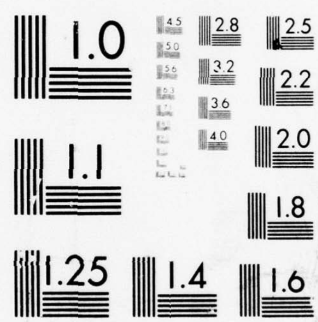
```
      N3=N2+NUMNP
      N4=N3+NUMNP
      N5=N4+NUMNP
      N6=N5+NUMNP
      N7=N6+NUMNP
      N8=N7+NUMEL
      N9=N8+NUMEL
      N10=N9+NUMEL
      N11=N10+NUMEL
      N12=N11+NUMEL
      N13=N12+NUMEL*3
      N14=N13+NUMEL*4
      N15=N14+NUMEL*4
      N16=N15+NEQ
      N17=N16+NEQ*MBAND
      N18=N17+NUMEL
      N19=N18+NUMEL
C
C
      CALL PRELIM(A(N1),A(N2),A(N3),A(N4),A(N5),A(N6))
C
C
      IF(N19 .LE. NFIELD) GO TO 100
      WRITE(6,1001) N19
      STOP
  100 CONTINUE
      WRITE(6,1002) N19
C
      CALL PLAST(A(N1),A(N2),A(N3),A(N4),A(N5),A(N6),A(N7),A(N8),A(N9),
     1A(N10),A(N11),A(N12),A(N13),A(N14),A(N15),A(N16),A(N17),A(N18),
     2NO,NEL,FLIMIT,ITCONT,ACCEF)
C
 1000 FORMAT(12A6)
 1001 FORMAT(///* THE DIMENSION OF THE ARRAY (A) IS TOO SMALL*/
     1* THE SIZE OF THE ARRAY (A) MUST BE *, I7)
 1002 FORMAT(//* THE NECCESSARY SIZE OF THE ARRAY (A) IS*, I7)
 1003 FORMAT(6I5,F10.0)
 1004 FORMAT(4F10.0)
 1005 FORMAT(4I5,F10.0)
C
      STOP
      END


      SUBROUTINE PRELIM(R,Z,UR,UZ,CODE,SLOP)
      COMMON/GENCON/NUMNP,NUMEL,HED(12),DLL,NEQ,NFORM,YIELD,TEST,ITER,
     1NREAD,NPUNCH,NPRINT,PVALUE,T,MBAND,RNFAC,RADIUS,PRES,DPRES
C
C
      DIMENSION R(1),Z(1),CODE(1),UR(1),UZ(1),SLOP(1)
C**********************************************************************
C     READ AND PRINT OF CONTROL INFORMATION AND MATERIAL PROPERTIES
C**********************************************************************
C
   50 CONTINUE
      WRITE (6,2000) HED,NUMNP,NUMEL
      CALL HARD(0.,YIELD)
      WRITE(6,2010) YIELD
      WRITE(6,1009) ITER
C
C**********************************************************************
C     READ AND PRINT OF NODAL POINT DATA
C**********************************************************************
C
      L=0
      IF(NPRINT.EQ.0) GO TO 60
      WRITE (6,1114)
      WRITE (6,2004)
   60 READ  (5,1002) N,CODE(N),R(N),Z(N),UR(N),UZ(N),SLOP(N)
      NL=L+1
      ZX=N-L
      IF(L .EQ. 0) GO TO 70
      DR=(R(N)-R(L))/ZX
      DZ=(Z(N)-Z(L))/ZX
      DUR=(UR(N)-UR(L))/ZX
      DUZ=(UZ(N)-UZ(L))/ZX
      DS=(SLOP(N)-SLOP(L))/ZX
C
   70 L=L+1
      IF(N-L) 100,90,80
C
```

AD-A068 250 CALIFORNIA UNIV BERKELEY DEPT OF MECHANICAL ENGINEERING F/G 13/8
ANALYSIS OF AXISYMMETRIC SHEET-METAL FORMING PROCESSES BY THE R--ETC(U)
SEP 78 F33615-77-C-5111

UNCLASSIFIED AFML-TR-78-120 NL

2 OF 2
AD
A068250

END
DATE
FILMED
6--79
DDC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

```
   80 CODE(L)=0.0                                                       MAIN0043
      R(L)=R(L-1)+DR                                                    MAIN0044
      Z(L)=Z(L-1)+DZ
      UR(L)=UR(L-1)+DUR
      UZ(L)=UZ(L-1)+DUZ
      SLOP(L)=SLOP(L-1)+DS
      GO TO 70                                                          MAIN0045
   90 IF(NUMNP-N) 100,110,60
C
  100 WRITE (6,2009) N                                                  MAIN0052
      CALL EXIT                                                         MAIN0053
C
  110 CONTINUE                                                          MAIN0054
      WRITE (6,2002) (K,CODE(K),R(K),Z(K),UR(K),UZ(K),SLOP(K),K=1,NUMNP)
C
      NEQ=3*NUMNP
      WRITE(6,1122) NEQ,MBAND
C***********************************************************************MAIN0119
 1002 FORMAT (I5,F5.0,5F10.0)                                           MAIN0122
 1003 FORMAT(16I5)
 1004 FORMAT(I8,2I11,2F10.6)
 1005 FORMAT(2I5,4F10.0)
 1006 FORMAT(// * THE NODAL POINTS AT WHICH FORCE CALCULATIONS ARE DESIR
     1ED* // 20I5)
 1007 FORMAT(1H1,15X, 39H LINEARLY DISTRIBUTED BOUNDARY STRESSES/
     1 / 60H    NODE I    NODE J   PRESSURE I   PRESSURE J       SHEAR I
     2 14H         SHEAR J)
 1008 FORMAT(2I9,4E15.5)
 1009 FORMAT(///* MAXIMUM NUMBER OF ITERATIONS ALLOWED FOR EACH INCREMEN
     1T =*, I3)
 1114 FORMAT(1H1, 39H NODAL POINT INFORMATION BEFORE SCALING//)
 1122 FORMAT(/// * NUMBER OF EQUATIONS  =*, I4/
     1            * BANDWIDTH               =*, I4/
     2            * DIAGONAL ELEMENTS     =*, I4 )
 2000 FORMAT (1H  12A6/
     1 30H0 NUMBER OF NODAL POINTS------ I3 /                           MAIN0127
     2 30H0 NUMBER OF ELEMENTS---------- I3 /)
 2002 FORMAT (I12,F12.2,2F12.3,3E24.7)
 2003 FORMAT (1I13,4I6,1I12)                                            MAIN0137
 2004 FORMAT (/  * NODAL POINT        TYPE  R-ORDINATE  Z-ORDINATE  R LO
     1AD OR DISPLACEMENT  Z LOAD OR DISPLACEMENT          BETA-SLOPE *)
 2005 FORMAT(//,*FORCES SPECIFIED AT NODAL POINT*,//,
     1 * NODAL PT.   ELEMENT1   ELEMENT2    PRESSURE       SHEAR*,/)
 2009 FORMAT (26H0NODAL POINT CARD ERROR N= I5)                         MAIN0145
 2010 FORMAT(// * INITIAL YIELD STRESS = *, F15.7//)
      RETURN
      END


      SUBROUTINE PLAST(R,Z,UR,UZ,CODE,SLOP,YY,YX,SPHI,CPHI,DL,STS,TEPS,
     1EPS,B,A,THICK,PHI,NQ,NEL,FLIMIT,ITCONT,ACOEF)
C
C***********************************************************************
C     PLAST IS THE CONTROLLING SUBROUTINE
C***********************************************************************
C
      COMMON/GENCON/NUMNP,NUMEL,HED(12),DLL,NEQ,NFORM,YIELD,TEST,ITER,
     1NREAD,NPUNCH,NPRINT,RVALUE,T,MBAND,PNRAD,RADIUS,PRES,DPRES
      COMMON/CONQUAD/SS(4),WT(4),D(2,2),SQRT1
      COMMON/FORVE/FACT,DFACT
C
      DIMENSION R(1),Z(1),UR(1),UZ(1),CODE(1),SLOP(1),YY(1),YX(1),
     1TEPS(4,1),B(1),A(NQ,1),THICK(1),CPHI(1),SPHI(1),DL(1),EPS(4,1),
     2PHI(1),STS(3,1)
C
C
      DFACT=DPRES
      FACT=PRES
C
      SS(1)=0.8611363116
      SS(2)=0.3399810436
      SS(3)=-SS(1)
      SS(4)=-SS(2)
C
      DO 442 N=1, NUMEL
      THICK(N)=T
      DO 442 I=1, 4
      TEPS(I,N)=0.
      TEPS(4,N)=0.0001
  442 CONTINUE
C
C
```

```
C*****************************************************************************
C     IF THE COMPUTATION IS INTERRUPTED AFTER A NUMBER OF STEPS
C     AND RESTARTED, THEN NECESSARY DATA NEED BE FEED
C*****************************************************************************
C
      IF(NREAD .LE. 0) GO TO 440
      READ(5,1017) (UR(I), UZ(I), SLOP(I), I=1,NUMNP)
      IF(ITCONT .EQ. 1) GO TO 440
      READ(5,1017) ( R(I),  Z(I), I=1, NUMNP)
      READ(5,1017)((TEPS(I,N),I=1,4), N=1, NUMEL)
      READ(5,1017)(THICK(N),N=1,NUMEL)
      READ(5,2223)FACT
  440 CONTINUE
C
C
      NSTEP=0
 2100 NSTEP=NSTEP+1
C
C
C*****************************************************************************
C     COMPUTE THE YIELD STRESS AND THE WORKHARDENING RATE
C*****************************************************************************
C
      DO 220 N=1, NUMEL
      CALL HARD(TEPS(4,N),YY(N))
      CALL HARD2(TEPS(4,N),YX(N))
  220 CONTINUE
C
C
C
      WRITE(6,1007) NSTEP
C
C
  659 CONTINUE
C
C*****************************************************************************
C     DETAIL OF THE PRESENT CONFIGURATION
C     SPHI=SINE OF ANGLE PHI
C     CPHI=COSINE OF ANGLE PHI
C     DL=ELEMENT LENGTH
C*****************************************************************************
C
      WRITE(6,1031)
      DO 690 N=1, NUMEL
      NP1=N+1
      DR=R(N)-R(NP1)
      DZ=Z(NP1)-Z(N)
      DL(N)=SQRT(DR*DR+DZ*DZ)
      SPHI(N)=DR/DL(N)
      CPHI(N)=DZ/DL(N)
      PHI(N)=ASIN(SPHI(N))*180./3.14156
      WRITE(6,1030)N,PHI(N),THICK(N),DL(N)
  690 CONTINUE
C
      RP1=RVALUE+1.
      RCONST=3.*RP1/( 2.*(1.+RVALUE+RVALUE) )
      D(1,1)=RP1*RCONST
      D(1,2)=RVALUE*RCONST
      D(2,1)=D(1,2)
      D(2,2)=D(1,1)
      CLAMDA=RP1*RP1/(1.+RVALUE+RVALUE)
      POISON=RVALUE/RP1
C
      DO 2000 K=1, ITER
C
C
      CALL STIFF(R,Z,UR,UZ,CODE,SLOP,YY,YX,SPHI,CPHI,DL,EPS,
     1THICK,A,B,NQ)
C
C
C*****************************************************************************
C     INTRODUCTION OF BOUNDARY CONDITION
C*****************************************************************************
C
      CALL MODIFY(CODE,A,B,NUMNP,NEQ,MBAND)
C
C*****************************************************************************
C     BANDED SYMMETRIC SOLUTION
C*****************************************************************************
C
      CALL TRIA(NEQ,MBAND,A)
      CALL BACKS(NEQ,MBAND,A,B)
C
C
C
```

```
      DO 130 I=1, NUMNP
      IZ=3*I-1
      IR=IZ-1
      IS=IZ+1
      UR(I)=UR(I)+B(IR)*ACCEF
      UZ(I)=UZ(I)+B(IZ)*ACCEF
      SLOP(I)=SLOP(I)+B(IS)*ACCEF
  130 CONTINUE
C
C
      WRITE(6,1016) K
      WRITE(6,1006) K
      DO 776 I=1, NUMNP
      IZ=3*I-1
      IR=IZ-1
      IS=IZ+1
      WRITE(6,1002) I,B(IR),B(IZ),B(IS),UR(I),UZ(I),SLOP(I),R(I),Z(I)
  776 CONTINUE
C
C
C
C***********************************************************************
C     COMPUTE NORM OF ERROR AND NORM OF SOLUTION.
C***********************************************************************
      ENORM = 0.
      SNORM = 0.
      DO 134 I=1, NUMNP
      IZ=3*I-1
      IR=IZ-1
      IS=IZ+1
      ENORM = ENORM + B(IR)*B(IR) + B(IZ)*B(IZ) + B(IS)*B(IS)
      SNORM = SNORM + UR(I)*UR(I) + UZ(I)*UZ(I) + SLOP(I)*SLOP(I)
  134 CONTINUE
      ENORM = SQRT(ENORM)
      SNORM = SQRT(SNORM)
      ESNORM=ENORM/SNORM
      WRITE(6,1015) SNORM,ENORM,ESNORM
C
C
  131 CONTINUE
C
C
C
C***********************************************************************
C     COMPUTE STRAIN FROM THE NEW GUESS.
C     EPS(1,N)=INCREMENT OF MERIDIAN STRAIN
C     EPS(2,N)=INCREMENT OF TANGENTIAL STRAIN
C     EPS(3,N)=INCREMENT OF THICKNESS STRAIN
C***********************************************************************
C
C
      DO 800 N=1,NUMEL
      NP1=N+1
      DLL=DL(N)
      SPH=SPHI(N)
      CPH=CPHI(N)
      AU=UR(N)+UR(NP1)
      AR=R(N)+R(NP1)
      DR=R(N)-R(NP1)
      DZ=Z(NP1)-Z(N)
      DU=UR(N)-UR(NP1)
      DW=UZ(NP1)-UZ(N)
      EX1=1.+2.*DR*DU/DLL/DLL+2.*DZ*DW/DLL/DLL+(DU*DU+DW*DW)/DLL/DLL
      EPS(1,N)=SQRT(EX1)-1.
      EPS(1,N)=ALOG(1.+EPS(1,N))
      EPS(2,N)=AU/AR
      EPS(2,N)=ALOG(1.+AU/AR)
      EPS(3,N)=-EPS(1,N)-EPS(2,N)
C
  800 CONTINUE
C
C
      TEST=0.0
      WRITE(6,1026) NSTEP
C
C
C
C***********************************************************************
C     COMPUTE INCREMENT OF EFFECTIVE STRAIN
C***********************************************************************
C
      DO 222 N=1, NUMEL
      ES=EPS(1,N)
      ET=EPS(2,N)
      RBAR=RR1*(ES*ES+ET*ET) + 2.*RVALUE*ES*ET
```

```
      EPS(4,N)=SQRT(2.*RCCNST*RPAR/3.)
C
C
C
C*****************************************************************************
C     COMPUTE STRESS DISTRIBUTION
C     STS(1,N)=MERIDIAN STRESS
C     STS(2,N)=CIRCUMFERENTIAL STRESS
C     STS(3,N)=EFFECTIVE STRESS
C*****************************************************************************
C
      STS(1,N)=CLAMDA*(ES + POISON*ET)*YY(N)/EPS(4,N)
      STS(2,N)=CLAMDA*(ET + POISON*ES)*YY(N)/EPS(4,N)
      IF(ESNORM .LT. FLIMIT)TEST=1.0
      DO 443 I=1, 4
  443 TEPS(I,N)=TEPS(I,N)+EPS(I,N)*TEST
      ES=STS(1,N)
      ET=STS(2,N)
      EFSTRS=ES*ES+ET*ET-2.*POISON*ES*ET
      STS(3,N)= SQRT(EFSTRS)
C
C
      WRITE(6,1003)N,EPS(1,N),TEPS(1,N),EPS(2,N),TEPS(2,N),EPS(3,N),TEP
     1S(3,N),EPS(4,N),TEPS(4,N)
      IF(ESNORM .LT. FLIMIT)THICK(N)=THICK(N)*EXP(EPS(3,N))
  222 CONTINUE
      WRITE(6,1027)
C
      DO 430 N=1,NUMEL
  430 WRITE(6,2251)N,(STS(I,N),I=1,3)
C
C
      IF(ESNORM .LT. FLIMIT)FACT=FACT+DFACT
      IF(ESNORM .LT. FLIMIT)WRITE(6,1028)FACT
      IF(ESNORM .LT. FLIMIT)GO TO 438
C
 2000 CONTINUE
 2200 CONTINUE
C
C
  438 CONTINUE
C
C
C*****************************************************************************
C     NEW CONFIGURATION
C*****************************************************************************
C
      DO 439 I=1, NUMNP
      IZ=3*I-1
      IR=IZ-1
      R(I)=R(I)+UR(I)*TEST
      Z(I)=Z(I)+UZ(I)*TEST
  439 CONTINUE
  777 CONTINUE
C
C
C
      IF(NPUNCH .EQ. 0) GO TO 310
      PUNCH 1017, (UR(I),UZ(I), SLOP(I), I=1,NUMNP)
      PUNCH 1017, (R(I),Z(I),I=1, NUMNP)
      PUNCH 1017, ( (TEPS(I,N),I=1,4), N=1, NUMEL)
      PUNCH 1017,(THICK(N),N=1,NUMEL)
      PUNCH 2223,FACT
  310 CONTINUE
C
      IF(ESNORM .GT. FLIMIT)GO TO 2300
C
      IF(NSTEP .LT. NFORM) GO TO 2100
C
 2300 CONTINUE
  500 CONTINUE
C
 1002 FORMAT(I5,3F13.7,5X,3F13.7,5X,3F13.7)
 1003 FORMAT(I7,11F11.6)
 1004 FORMAT(16I5)
 1005 FORMAT(1H1,* STRAIN-STRESS SOLUTION AT  STEP  NUMBER  =*,I4//
     1 * EL. NO...R-STRAIN...Z-STRAIN..TH-STRAIN...RZ-STRAIN...EF-STRAIN
     2...R-STRES...Z-STRES...TH-STRES...RZ-STRES..EF-STRES..AVG-STRES..*
     3)
 1006 FORMAT(///  30X, * DISPLACEMENT SOLUTION AT ITERATION NUMBER =*,I4
     1///  20X, * PERTURBED*, 26X, * TOTAL*, 20X, * DEFORMED COORD*/
     2/   *   NP         DU            DW             DBETA          U
     3      W                   BETA                R              Z*)
 1007 FORMAT(1H1,70X,*ITERATION PROCESS FOR STEP*,I4)
 1008 FORMAT(      60X, * TOTAL R-LOAD =*, F12.7
```

```
     1            /  60X, * TOTAL Z-LOAD =*, F12.7
     2            /  60X, * TOTAL W-LOAD =*, F12.7)
 1010 FORMAT(    // * NODAL POINT FORCE AT STEP =*, I4//
     1*.......N.F.........R-FORCE.......Z-FORCE......Z-STRESS ON DIE SUR
     2FACE...*)
 1011 FORMAT(I5,3F10.0)
 1012 FORMAT(I9,6F17.5)
 1015 FORMAT(60X,* VELOCITY CONVERGENCE*,/
     1               60X, * NORM OF SOLUTION VECTOR =*, F13.8
     1            /  60X, * NORM OF    ERROR  VECTOR =*, F13.8
     2            /  60X, * FRACTIONAL  NORM         =*, F13.8)
 1016 FORMAT(            * DISPLACEMENT SOLUTION AT ITERATION NUMBER =*,I4)
 1017 FORMAT(8F10.7)
 1018 FORMAT(////* DOES NOT CONVERGE*//
     1* TRY AGAIN WITH DECELLERATION COEFFICIENT =ACOEF= LESS THAN*,
     2F8.3)
 1020 FORMAT(20F4.1)
 1025 FORMAT(4X,I5,3X,F12.6,10X,I5,3X,F12.6,10X,I5,3X,F12.6)
 2251 FORMAT(I5,4F20.7)
 1026 FORMAT(///*INCREMENTAL STRAIN-TOTAL STRAIN AT STEP NUMBER=*,I4//
     1*EL  NO....S-STRAIN......TOTAL...THE-STRAIN......TOTAL....THI-STR
     2AIN......TOTAL....EF-STRAIN......TOTAL...*)
 1027 FORMAT(///*EL.  NO....S-STRESS....THE-STRESS....EF-STRESS....*)
 1030 FORMAT(/I7,3F10.5)
 1031 FORMAT(*  GEOMETRY  OF  PROFILE*//
     1 *EL  NNO.....ANGLE.....THICKNESS.....*)
 1028 FORMAT(*     FORCE AT THIS  STEP IS *,F20.7)
 2223 FORMAT(F20.7)
C
      RETURN
      END
```

```
      SUBROUTINE STIFF(R,Z,UR,UZ,CODE,SLOP,VY,VX,SPHI,CPHI,DL,EPS,
     1THICK,A,B,NQ)
C
      COMMON/GENCON/NUMNP,NUMEL,HED(12),DLL,NEQ,NFORM,YIELD,TEST,ITER,
     1NREAD,NPUNCH,NPRINT,RVALUE,T,MBAND,PNRAD,RADIUS,PRES,DPRES
      COMMON/STFMAT/H(6),P(6,6),TEX,TEY,TEZ,THKL
      COMMON/CONQUAD/SS(4),WT(4),D(2,2),SQRT1
C
      DIMENSION R(1),Z(1),CODE(1),UR(1),UZ(1),SLOP(1),B(1),A(NQ,1),
     1 EPS(4,1),ZZ(2),UU(6),VY(1),VX(1),THICK(1),DL(1),SPHI(1),CPHI(1)
     2,RR(2)
C
C
      DO 50 N=1, NEQ
      B(N)=0.
      DO 50 M=1,MBAND
   50 A(N,M)=0.
C
      WT(1)=0.3478548451
      WT(2)=0.6521451549
      WT(3)=WT(1)
      WT(4)=WT(2)
C
C
      DO 1000 N=1, NUMEL
      NP1=N+1
      DLL=DL(N)
      SPH=SPHI(N)
      CPH=CPHI(N)
      RR(1)=R(N)
      ZZ(1)=Z(N)
      RR(2)=R(NP1)
      UU(1)=UR(N)
      UU(2)=UZ(N)
      UU(3)=SLOP(N)
      UU(4)=UR(NP1)
      UU(5)=UZ(NP1)
      UU(6)=SLOP(NP1)
      THKL=THICK(N)*DLL
      ZZ(2)=Z(NP1)
      VG=VX(N)
      VH=VY(N)
C
      CALL QUAD(RR,ZZ,UU,DLL,SPH,CPH,VG,VH)
C
C
C
C*****************************************************************************
C     PERFORM THE ASSEMBLY OPERATION. BECAUSE MATRIX A IS SYMMETRIC
```

```
C     ONLY UPPER HALF OF THE MATRIX IS CREATED.  AND THE STORAGE FOR
C     MATRIX A IS A SQUARE ARRAY BECAUSE OF BANDED SYMMETRIC PROPERTY
C**********************************************************************
C
      DO 200 I=1, 6
      II=N*3 - 3 + I
      B(II)=B(II)+H(I)
      DO 200 J=1, 6
C
C
      JJ= N*3 - 3 + J - II + 1
      IF(JJ .LT. 1) GO TO 200
      A(II,JJ)=A(II,JJ)+P(I,J)
  200 CONTINUE
C
 1000 CONTINUE
C
C
 1001 FORMAT(///,* THE DIAGONAL VECTOR OF MATRIX OF STIFFNESS*/)
 1002 FORMAT(12E11.3)
 1005 FORMAT(// 29H ELEMENT WITH NEGATIVE AREA =, I5)
C
      RETURN
      END
```

```
      SUBROUTINE QUAD(RR,ZZ,UU,DLL,SPH,CPH,S2,S1)
C
      COMMON/ISOTRY/RVAL1
      COMMON/STFMAT/H(6),P(6,6),TEX,TEY,TEZ,THKL
      COMMON/CONQUAD/SS(4),WT(4),D(2,2),SQRT1
      COMMON/FORVE/FACT,DFACT
C
      DIMENSION RR(2),ZZ(2),UU(6),P(2,6),XX(6,6),BZERO(6),DR(2,6)
      DIMENSION RA(6,6),RB(6)
C
C
      RC=(RR(1)+RR(2))/2.
C
      DO 2 I=1,6
      RB(I)=0.
      H(I)=0.
      DO 2 J=1,6
      RA(I,J)=0.
    2 P(I,J)=0.
C
C
C
      RVALLE=RVAL1
      DZ=ZZ(2)-ZZ(1)
      DR=RR(1)-RR(2)
      DU=UU(1)-UU(4)
      DW=UU(5)-UU(2)
      AU=UU(1)+UU(4)
      AR=RR(1)+RR(2)
C
C
      C1=2.*DR/DLL/DLL
      C2=2.*DU/DLL/DLL
      C3=2.*DZ/DLL/DLL
      C4=2.*DW/DLL/DLL
      C5=AU/AR/2.
      C6=1.+DR*C2+DZ*C4+(DU*DU+DW*DW)/DLL/DLL
      C7=2./DLL/DLL
      C8=2./AR/AR
      C9=1./SQRT(C6)/2.
      C10=C9/C6
      C11=C1+C2
      C12=C3+C4
C
C
      DES1=SQRT(C6)
      DET1=2.*C5+1.
C
      E1=C9*C11/DES1
      E2=-C9*C12/DES1
      E3=-E1
      E4=-E2
      E5=1./AR/DET1
      E6=(-C10*C11*C11/2.+C9*C7)/DES1-E1*E1
      E7=-E6
```

```
      E8=C10*C11*C12/2./DES1-E1*E2
      E9=-E8
      E10=E8
      E11=(-C10*C12*C12/2.+C9*C7)/DES1-E2*E2
      E12=-E5*E5
C
C
      DES=ALOG(DES1)
      DET=ALOG(DET1)
C***********************************************************************
C     DES=MERIDIAN STRAIN INCREMENT
C     DET=CIRCUMFERENTIAL STRAIN INCREMENT
C     COMPUTATION OF EFFECTIVE STRAIN INCREMENT
C     E1=DERIVATIVE OF MERIDIAN STRAIN INCREMENT WITH RESPECT TO UU(1)
C       =D(DES)/D(UU(1))
C     E2=D(DES)/D(UU(2))
C     E3=D(DES)/D(UU(4))
C     E4=D(DES)/D(UU(5))
C     E5=D(DET)/D(UU(1))
C     E6=D(E1)/D(UU(1))
C     E7=D(E1)/D(UU(4))
C     E8=D(E1)/D(UU(2))
C     E9=D(E3)/D(UU(2))
C     E10=D(E4)/D(UU(5))
C     E11=D(E5)/D(UU(2))
C***********************************************************************
C
C
      RVP1=RVALUE+1.
      RVP2=SQRT(2.*RVALUE+1.)
      RVP3=RVP1/RVP2
      RVP4=2.*RVALUE/RVP1
C
C*******************************************************************
C     EFFECTIVE  STRAIN
C*******************************************************************
C
      EFS=DES*DES+DET*DET+RVP4*DES*DET
      EFS1=RVP3*SQRT(EFS)
      EFS2=RVP3/SQRT(EFS)/2.
      EFS3=-RVP3/EFS/SQRT(EFS)/4.
      D1=(2.*DES+RVP4*DET)*E1+(2.*DET+RVP4*DES)*E5
      D2=(2.*DES+RVP4*DET)*E2
      D3=(2.*DES+RVP4*DET)*E3+(2.*DET+RVP4*DES)*E5
      D4=(2.*DES+RVP4*DET)*E4
C
      F1=EFS2*D1
      F2=EFS2*D2
      F3=EFS2*D3
      F4=EFS2*D4.
      F11=EFS3*D1*D1+EFS2*((2.*DES+RVP4*DET)*E6+(2.*DET+RVPA*DES)*E12
     1+(2.*E1+RVP4*E5)*E1+(2.*E5+RVP4*E1)*E5)
      F12=EFS3*D1*D2+EFS2*((2.*DES+RVP4*DET)*E8+(2.*E1+RVP4*E5)*E2)
      F13=EFS3*D1*D3+EFS2*((2.*DES+RVP4*DET)*E7+(2.*E3+RVP4*E5)*E1
     1+(2.*E5+RVP4*E3)*E5+(2.*DET+RVP4*DES)*E12)
      F14=EFS3*D1*D4+EFS2*((2.*DES+RVP4*DET)*E9+(2.*E1+RVP4*E5)*E4)
      F22=EFS3*D2*D2+EFS2*((2.*DES+RVP4*DET)*E11+2.*E2*E2)
      F23=EFS3*D2*D3+EFS2*((2.*DES+RVP4*DET)*E9+(2.*E3+RVP4*E5)*E2)
      F24=EFS3*D2*D4+EFS2*(-(2.*DES+RVP4*DET)*E11+2.*E4*E2)
      F33=EFS3*D3*D3+EFS2*((2.*DES+RVPA*DET)*E6+(2.*E3+RVP4*E5)*E3+(
     12.*E5+RVP4*E3)*E5+(2.*DET+RVP4*DES)*E12)
      F34=EFS3*D3*D4+EFS2*((2.*DES+RVP4*DET)*E10+(2.*E3+RVPA*E5)*E4)
      F44=EFS3*D4*D4+EFS2*((2.*DES+RVP4*DET)*E11+2.*E4*E4)
C
C
C
C*******************************************************************
C     F1=DERIVATIVE OF EFFECTIVE STRAIN INCREMENT WITH RESPECT TO UU(1)
C     F2=WITH RESPECT TO UU(2)
C     F3=WITH RESPECT TO UU(4)
C     F4=WITH RESPECT TO UU(5)
C     F11=D(F1)/D(UU(1))
C     F12=D(F1)/D(UU(2))
C     F13=D(F1)/D(UU(4))
C     F14=D(F1)/D(UU(5))
C     F22=D(F2)/D(UU(2))
C     F23=D(F2)/D(UU(4))
C     F24=D(F2)/D(UU(5))
C     F33=D(F3)/D(UU(4))
C     F34=D(F3)/D(UU(5))
C     F44=D(F4)/D(UU(5))
C*******************************************************************
C
      P(1,1)=((S1+S2*EFS1)*F11+S2*F1*F1)*RC*THKL
      P(1,2)=((S1+S2*EFS1)*F12+S2*F2*F1)*RC*THKL
```

```
      P(1,4)=((S1+S2*EFS1)*F13+S2*F1*F3)*RC*THKL
      P(1,5)=((S1+S2*EFS1)*F14+S2*F1*F4)*RC*THKL
      P(2,2)=((S1+S2*EFS1)*F22+S2*F2*F2)*RC*THKL
      P(2,4)=((S1+S2*EFS1)*F23+S2*F2*F3)*RC*THKL
      P(2,5)=((S1+S2*EFS1)*F24+S2*F2*F4)*RC*THKL
      P(4,4)=((S1+S2*EFS1)*F33+S2*F3*F3)*RC*THKL
      P(4,5)=((S1+S2*EFS1)*F34+S2*F3*F4)*RC*THKL
      P(5,5)=((S1+S2*EFS1)*F44+S2*F4*F4)*RC*THKL
      P(2,1)=P(1,2)
      P(4,1)=P(1,4)
      P(4,2)=P(2,4)
      P(5,1)=P(1,5)
      P(5,2)=P(2,5)
      P(5,4)=P(4,5)
C
      H(1)=-(S1+S2*EFS1)*F1*RC*THKL
      H(2)=-(S1+S2*EFS1)*F2*RC*THKL
      H(4)=-(S1+S2*EFS1)*F3*RC*THKL
      H(5)=-(S1+S2*EFS1)*F4*RC*THKL
C
C
C*****************************************************************************
C   ENCLOSED   VOLUME   CHANGE
C*****************************************************************************
C
      RK1=DLL*CPH+DW
      RUU1=RR(1)+UU(1)
      RUU2=RR(2)+UU(4)
      RK2=2.*RUU1+RUU2
      RK3=2.*RUU2+RUU1
      RA(1,1)=RK1/3.
      RA(1,2)=-RK2/6.
      RA(1,4)=RK1/6.
      RA(1,5)=RK2/6.
      RA(2,2)=0.
      RA(2,4)=-RK3/6.
      RA(2,5)=0.
      RA(4,4)=RK1/3.
      RA(4,5)=RK3/6.
      RA(5,5)=0.
      RB(1)=RK2*RK1/6.
      RB(4)=RK3*RK1/6.
      RB(2)=-(RUU1*RUU1+RUU2*RUU2+RUU1*RUU2)/6.
      RB(5)=-RB(2)
      RA(2,1)=RA(1,2)
      RA(4,1)=RA(1,4)
      RA(4,2)=RA(2,4)
      RA(5,1)=RA(1,5)
      RA(5,2)=RA(2,5)
      RA(5,4)=RA(4,5)
C
C
      DO 108 I=1,6
      H(I)=H(I)+(FACT+DFACT)*RB(I)
      DO 108 J=1,6
      P(I,J)=P(I,J)-(FACT+DFACT)*RA(I,J)
  108 CONTINUE
C
C
   71 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE CONDEN(A,B,NEQ,MBAND,N,U)
C
C*****************************************************************************
C     PERFORM THE MATRIX CONDENSATION WHEN THE VALUE OF A COMPONENT
C     X IN AX=B IS SPECIFIED EQUAL TO ZERO
C*****************************************************************************
C
      DIMENSION B(NEQ),A(NEQ,1)
C
      DO 250 M=2,MBAND
      K=N-M+1
      IF(K) 235,235,230
  230 B(K)=B(K)-A(K,M)*U
      A(K,M)=0.0
  235 K=N+M-1
      IF(NEQ-K) 250,240,240
  240 B(K)=B(K)-A(N,M)*U
      A(N,M)=0.0
```

```
  250 CONTINUE
      A(N,1)=1.0
      B(N)=U
C
      RETURN
      END
```

```
      SUBROUTINE MODIFY(CODE,A,B,NUMNP,NEQ,MBAND)
C
      DIMENSION CODE(1),A(NEQ,1),B(1)
C
C
      DO 121 I=1, NUMNP
      IL=3*I
      IZ=IL-1
      IR=IZ-1
      C=CODE(I)
      IF (C.EQ. 1.) GO TO 101
      IF (C.EQ. 2.) GO TO 102
      IF (C.EQ. 3.) GO TO 103
      CALL CONDEN(A,B,NEQ,MBAND,IL,0.)
      GO TO 121
C
  101 CONTINUE
      CALL CONDEN(A,B,NEQ,MBAND,IR,0.)
      CALL CONDEN(A,B,NEQ,MBAND,IL,0.)
      GO TO 121
C
  102 CONTINUE
      CALL CONDEN(A,B,NEQ,MBAND,IZ,0.)
      CALL CONDEN(A,B,NEQ,MBAND,IL,0.)
      GO TO 121
C
  103 CONTINUE
      CALL CONDEN(A,B,NEQ,MBAND,IR,0.)
      CALL CONDEN(A,B,NEQ,MBAND,IZ,0.)
      CALL CONDEN(A,B,NEQ,MBAND,IL,0.)
  121 CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE TRIA(NN,MM,A)
C
C**************************************************************************
C     TRIANGULIZATION OF GAUSSIAN ELIMINATION FOR THE SOLUTION
C     OF BANDED SYMMETRIC MATRIX
C**************************************************************************
C
      DIMENSION A(NN,1)
C
      N=0
  100 N=N+1
      IF(N.EQ.NN)RETURN
      IF(A(N,1).NE.0.) GO TO 150
      GO TO 100
C
  150 I=N
      MR=MINO(MM,NN-N+1)
C
      DO 260 L=2,MR
      I=I+1
      C=A(N,L)/A(N,1)
      IF(C.EQ.0.0)GO TO 260
      J=0
      DO 250 K=L,MR
      J=J+1
  250 A(I,J)=A(I,J)-C*A(N,K)
      A(N,L)=C
  260 CONTINUE
C
      GO TO 100
C
      END
```

```
      SUBROUTINE BACKS(NN,MM,A,B)
C
C
C***********************************************************************
C     BACK SUBSTITUTION FOR SOLUTION OF BANDED SYMMETRIC MATRIX
C***********************************************************************
C
      DIMENSION A(1),B(1)
C
      MMM=MM-1
      N=0
  270 N=N+1
      C=B(N)
      IF(A(N).NE.0.0)B(N)=B(N)/A(N)
      IF(N.EQ.NN)GO TO 300
      IL=N+1
      IH=MINO(NN,N+MMM)
      M=N
      DO 285 I=IL,IH
      M=M+NN
  285 B(I)=B(I)-A(M)*C
      GO TO 270
C
  300 IL=N
      N=N-1
      IF(N.EQ.0) RETURN
      IH=MINO(NN,N+MMM)
      M=N
      DO 400 I=IL,IH
      M=M+NN
  400 B(N)=B(N)-A(M)*B(I)
      GO TO 300
C
      END
```

```
      SUBROUTINE HARD(EPS,Y)
C
C***********************************************************************
C     WORKHARDENING CHARACTERISTIC CURVE
C***********************************************************************
C
      COMMON/MATERL/YVALUE,PRESTN,EXPNT,PRESTS
C
      Y=YVALUE*(PRESTN+EPS)**EXPNT+PRESTS
C
      RETURN
      END
```

```
      SUBROUTINE HARD2(EPS,Y)
C
C***********************************************************************
C     COMPUTE WORK HARDENING RATE
C***********************************************************************
C
      COMMON/MATERL/YVALUE,PRESTN,EXPNT,PRESTS
C
      Y=EXPNT*YVALUE*(PRESTN+EPS)**(EXPNT-1.)
C
      RETURN
      END
```

APPENDIX C

PROGRAM FOR THE ANALYSIS OF PUNCH STRETCHING

This program is for the analysis of the stretching of a sheet with hemispherical punch, where the die profile is neglected.

(I) Data card preparation

1. Read HED (A 12)
2. Read RVALUE, T, ACOEF (5F 10.0)
3. Read ITER, NREAD, ITCONT, NFORM, NPUNCH, NPRINT, FLIMIT (6I5, F10.0)
4. Read NUMNP (6I5)
5. Read PNRAD, RADIUS, FRITN (4F 10.0)

   PNRAD: Radius of the hemispherical punch

   RADIUS: Radius of the blank

   FRITN: Friction coefficient between the punch head and the blank
6. Read YVALUE, PRESTN, EXPNT, PRESTS (4F 10.0)
7. Read ECONST, TDIST (4F 10.0)

   ECONST: Step size in terms of the maximum magnitude of the effective strain increment. To start with, set this 0.04

   TDIST: Criterion distance of the contact of the sheet with the punch head. To start with, set this 0.008
8. Read N, CODE(N), R(N), Z(N), UR(N), UZ(N), SLOP(N), (I5, F5.0, 5F 10.0)

   Code (N) = 4.0 for the contact zone of the sheet with the punch head
9. If NREAD = 1, the new input data is to be placed behind the nodal information card

```
      PROGRAM STRCH(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,PUNCH)
C
C****************************************************************************
C     THIS PROGRAM IS FOR THE ANALYSIS OF THE PUNCH STRETCHING, BYJ.KIM
C     HERE, THE RADIUS OF THE DIE PROFILE IS NEGLECTED.
C****************************************************************************
C
      COMMON/GENCON/NUMNP,NUMEL,HED(12),DLL,NEQ,NFORM,YIELD,TEST,ITER,
     1NREAD,NPUNCH,NPRINT,RVALUE,T,MBAND,PNRAD,RADIUS,FRITN,
     2ECONST,ENHED,TDIST
      COMMON/MATERL/YVALUE,PRESTN,EXPNT,PRESTS
      COMMON/ISO/RRVAL
C
C****************************************************************************
C     PROGRAM        IS FOR CONTROLLING THE DIMENSION OF THE COMPLETE
C     PROGRAM.  ITS PURPOSE IS TO PREVENT ASSIGNING A LARGER THAN
C     NECESSARY DIMENSION FOR ANY ARRAY THROUGH THE USE OF THE
C     FOLLOWING STATEMENT
C****************************************************************************
C
      COMMON A(5000)
C
C****************************************************************************
C     NFIELD IS THE DIMENSION OF ARRAY A.  ITS VALUE CAN BE DETERMINED
C     PRECISELY BY RUNNING THE PROGRAM ONCE.
C****************************************************************************
C
      NFIELD=5000
C
C
C
C****************************************************************************
C     READ THE INPUT DATA CONTROL CARDS
C****************************************************************************
C
      TEST=1.
      READ(5,1000) HED
      READ(5,1004) RVALUE,T,ACOFF
      READ(5,1003) ITER,NREAD,ITCONT,NFORM,NPUNCH,NPRINT,FLIMIT
      READ(5,1003) NUMNP
      READ(5,1004) PNRAD,RADIUS,FRITN
      READ(5,1004) YVALUE,PRESTN,EXPNT,PRESTS
      READ(5,1004) ECONST,TDIST
C
C
C****************************************************************************
C     HED=OUTPUT TITLE
C     RVALUE=VALUE OF THE ANISOTROPY PARAMETER
C     ACOFF=ACCELERATING OR DECELERATING COEFFICIENT OF CONVERGENCE
C     NREAD=0, IF TO BYPASS THE READING STATEMENT IN SUBROUTINE PLAST
C     ITCONT=0, IF COMPUTATION STARTS AT THE VERY BEGGINNING AND FIRST/
C              SECOND STEPS ARE INCLUDED IN THE STEPS TO BE COMPUTED
C           =1, OTHERWISE
C            THIS INDEX IS RELATED TO THE DETERMINATION OF STEP SIZE
C     NFORM=NUMBER OF STEPS ASSIGNED PER RUN
C     NPUNCH=1, IF DATA ARE TO BE PUNCHED
C           =0, OTHERWISE
C     FLIMIT=VALUE OF (ERROR NORM)/(SOLUTION NORM) REQUIRED
C     FOR CONVERGENCE
C     NPRINT=1, IF NODAL POINT DATA ARE TO BE PRINTED
C           =0, OTHERWISE
C     NUMNP=NUMBER OF NODAL POINTS
C     PNRAD=RADIUS OF HEMISPHERICAL PUNCH HEAD
C     RADIUS=RADIUS OF THE BLANK
C     FRITN =FRICTION COEFFICIENT BETWEEN BLANK AND PUNCH
C     ECONST=STEP SIZE IN MAXIMUM EFFECTIVE STRAIN INCREMENT
C
C
C     YVALUE, PRESTN, EXPNT, PRESTS ARE TO EXPRESS THE WORKHARDENING
C     CHARACTERISTICS OF THE BLANK
C     STRESS=YVALUE*(PRESTN+STRAIN)**EXPNT+PRESTS
C
C     NEQ=NUMBER OF EQUATIONS TO BE SOLVED
C     NUMEL=NUMBER OF ELEMENTS
C     MBAND=BAND WIDTH
C****************************************************************************
C
```

```
C
      RRVAL=PVALUE
      NUMEL=NUMNP-1
      MBAND=6
      NEQ=NUMNP*3
      NQ=NEQ
      NEL=NUMEL
C
C
      N1=1
      N2=N1+NUMNP
      N3=N2+NUMNP
      N4=N3+NUMNP
      N5=N4+NUMNP
      N6=N5+NUMNP
      N7=N6+NUMNP
      N8=N7+NUMEL
      N9=N8+NUMEL
      N10=N9+NUMEL
      N11=N10+NUMEL
      N12=N11+NUMEL
      N13=N12+NUMEL*3
      N14=N13+NUMEL*4
      N15=N14+NUMEL*4
      N16=N15+NEQ
      N17=N16+NEQ*MBAND
      N18=N17+NUMEL
      N19=N18+NUMNP
      N20=N19+NUMNP
      N21=N20+NUMNP
      N22=N21+NUMNP
      N23=N22+NUMNP
      N24=N23+NEQ
      N25=N24+NUMNP
      N26=N25+NUMNP
      N27=N26+NUMNP
C
C
C
      IF(N27 .LE. NFIELD)GO TO 100
C
      WRITE(6,1001) N27
      STOP
C
  100 CONTINUE
      WRITE(6,1002) N27
C
C
      CALL PRELIM(A(N1),A(N2),A(N3),A(N4),A(N5),A(N6))
C
      CALL PLAST(A(N1),A(N2),A(N3),A(N4),A(N5),A(N6),A(N7),A(N8),A(N9),
     1A(N10),A(N11),A(N12),A(N13),A(N14),A(N15),A(N16),A(N17),A(N18),
     2A(N19),A(N20),A(N21),A(N22),A(N23),A(N24),A(N25),A(N26)
     3, NQ,NEL,FLIMIT,ITCONT,ACCEF)
C
 1000 FORMAT(12A6)
 1001 FORMAT(///* THE DIMENSION OF THE ARRAY (A) IS TOO SMALL*/
     1* THE SIZE OF THE ARRAY (A) MUST BE *, I7)
 1002 FORMAT(//* THE NECCESSARY SIZE OF THE ARRAY (A) IS*, I7)
 1003 FORMAT(6I5,F10.0)
 1004 FORMAT(4F10.0)
 1005 FORMAT(4I5,F10.0)
C
      STOP
      END
```

```
      SUBROUTINE PRELIM(R,Z,UR,UZ,CODE,SLOP)
C
C**************************************************************************MAINO
C     READ AND PRINT OF CONTROL INFORMATION AND MATERIAL PROPERTIES     MAIN0013
C**************************************************************************MAIN0014
C
C
      COMMON/GENCON/NUMNP,NUMEL,HED(12),CLL,NEQ,NFORM,YIELD,TEST,ITER,
     1NREAD,NPUNCH,NPRINT,PVALUE,T,MBAND,PNRAD,RADIUS,FRITN,
     2ECONST,FNHEC,TCIST
C
      DIMENSION R(1),Z(1),CODE(1),UR(1),UZ(1),SLOP(1)
C
C
   50 CONTINUE
```

```
      WRITE (6,2000) HED,NUMNP,NUMEL
      CALL HARD(0.,YIELD)
      WRITE(6,2010) YIELD
      WRITE(6,1009) ITER
C
C***********************************************************************MAIN0030
C     READ AND PRINT OF NODAL POINT DATA                                MAIN0031
C***********************************************************************MAIN0032
C
      L=0                                                               MAIN0034
      IF(NPRINT.EQ.0) GO TO 60
      WRITE (6,1114)
      WRITE (6,2004)                                                    MAIN0033
   60 READ  (5,1002) N,CODE(N),R(N),Z(N),UR(N),UZ(N),SLOP(N)
C
      NL=L+1                                                            MAIN0036
      ZX=N-L                                                            MAIN0037
      IF(L .EQ. 0) GO TO 70
      DR=(R(N)-R(L))/ZX                                                 MAIN0038
      DZ=(Z(N)-Z(L))/ZX                                                 MAIN0039
      DUR=(UR(N)-UR(L))/ZX
      DUZ=(UZ(N)-UZ(L))/ZX
      DS=(SLOP(N)-SLOP(L))/ZX
C
   70 L=L+1                                                             MAIN0041
      IF(N-L) 100,90,80                                                 MAIN0042
C
   80 CODE(L)=0.0                                                       MAIN0043
      R(L)=R(L-1)+DR                                                    MAIN0044
      SLOP(L)=SLOP(L-1)+DS
      Z(L)=Z(L-1)+DZ
      UR(L)=UR(L-1)+DUR
      UZ(L)=UZ(L-1)+DUZ
      GO TO 70                                                          MAIN0049
C
   90 IF(NUMNP-N) 100,110,60
C
  100 WRITE (6,2009) N                                                  MAIN0052
      CALL EXIT                                                         MAIN0053
C
  110 CONTINUE                                                          MAIN0054
      IF(NPRINT.EQ.0) GO TO 120
      WRITE (6,2002) (K,CODE(K),R(K),Z(K),UR(K),UZ(K),SLOP(K),K=1,NUMNP)
  120 CONTINUE
C
      NEQ=3*NUMNP
      WRITE(6,1122) NEQ,MBAND
C
 1002 FORMAT (I5,F5.0,5F10.0)                                           MAIN0122
 1003 FORMAT(16I5)
 1004 FORMAT(I8,2I11,2F10.6)
 1005 FORMAT(2I5,4F10.0)
 1006 FORMAT(// * THE NODAL POINTS AT WHICH FORCE CALCULATIONS ARE DESIR
     1ED* // 20I5)
 1007 FORMAT(1H1,15X, 39H LINEARLY DISTRIBUTED BOUNDARY STRESSES/
     1 / 60H    NODE I   NODE J   PRESSURE I   PRESSURE J       SHEAR I
     2 14H       SHEAR J)
 1008 FORMAT(2I9,4E15.5)
 1009 FORMAT(///* MAXIMUM NUMBER OF ITERATIONS ALLOWED FOR EACH INCREMEN
     1T =*, I3)
 1114 FORMAT(1H1, 39H NODAL POINT INFORMATION BEFORE SCALING//)
 1122 FORMAT(/// * NUMBER OF EQUATIONS  =*, I4/
     1               * BANDWIDTH               =*, I4/
     2               * DIAGONAL ELEMENTS       =*, I4 )
 2000 FORMAT (1H  12A6/
     1 30H0 NUMBER OF NODAL POINTS------ I3 /
     2 30H0 NUMBER OF ELEMENTS---------- I3 /)
 2002 FORMAT (I12,F12.2,2F12.3,3E24.7)
 2003 FORMAT (1I13,4I6,1I12)                                            MAIN0137
 2004 FORMAT (/  * NODAL POINT        TYPE  R-ORDINATE  Z-ORDINATE  R LO
     1AD OR DISPLACEMENT  Z LOAD OR DISPLACEMENT          BETA-SLOPE *)
 2005 FORMAT(//,*FORCES SPECIFIED AT NODAL POINT*,//,
     1 * NODAL PT.   ELEMENT1   ELEMENT2   PRESSURE        SHEAR*,/)
 2009 FORMAT (26H0NODAL POINT CARD ERROR N= I5)                         MAIN0145
 2010 FORMAT(// * INITIAL YIELD STRESS = *, F15.7//).
      RETURN
      END
```

```
      SUBROUTINE PLAST(R,Z,UR,UZ,CODE,SLOP,YY,YX,SPHI,CPHI,DL,STS,TEPS,
     1EPS,B,A,THICK,ALPHA,GAMMA,ETA,FRNFCE,PHI,FF,TOUCH,UUR,UUZ,
     2NQ,NEL,FLIMIT,ITCONT,ACOEF,NDEX)
```

```
      COMMON/GENCON/NUMNP,NUMEL,HED(12),DLL,NEG,NFORM,YIELD,TEST,ITER,
     1NREAD,NPUNCH,NPRINT,RVALUE,T,MBAND,PNHED,RADIUS,FRITN,
     2ECONST,FNHED,TDIST
C
C***********************************************************************
C     PLAST IS THE CONTROLLING SUBROUTINE
C***********************************************************************
C
      COMMON/CONQUAD/SS(4),WT(4),D(2,2),SQRT1
      DIMENSION R(1),Z(1),UR(1),UZ(1),CODE(1),SLOP(1),YY(1),YX(1),
     1TEPS(4,1),B(1),A(NQ,1),THICK(1),CPHI(1),SPHI(1),DL(1),EPS(4,1),
     2ALPHA(1),GAMMA(1),ETA(1),FRNFCE(1),PHI(1),FF(1),TOUCH(1)
     3,STS(3,1),UUR(1),UUZ(1)
      COMMON/ATOUCH/NTOUCH
C
C***********************************************************************
C     THE FIRST NODE IS LOCATED AT THE RIM OF THE BLANK
C     AND THE POLE IS THE LAST NODE
C***********************************************************************
C
C
C***********************************************************************
C     A1,A2 ARE CONSTANTS RELATED TO DETERMINATION OF ACOEF
C     PNHED=PRESENT POSITION OF PUNCH HEAD
C     NTOUCH=FIRST NODAL POINT IN CONTACT WITH PUNCH HEAD
C     NCHECK=NUMBER OF NODES FOR WHICH THE CONTACTING WILL BE CHECKED
C     TCHCOF=0, IF BOUNDARY IS TO ADVANCE
C           =1, OTHERWISE
C***********************************************************************
C
C
      PNHED=Z(NUMNP)
      A2=2.
      A1=10.
      NCHECK=NUMEL/10
      NUM1=NUMNP-1
      NTOUCH=NUMNP
      TCHCOF=0.0
      NSTOP=0
      SS(1)=0.8611363116
      SS(2)=0.3399810436
      SS(3)=-SS(1)
      SS(4)=-SS(2)
C
      DO 442 N=1, NUMEL
      FRNFCE(N)=0.0
      THICK(N)=T
      DO 442 I=1, 4
  442 TEPS(I,N)=0.
C***********************************************************************
C     TO HANDLE THE INFINITE WORKHARDENING RATE INITIALLY VERY
C     SMALL VALUE IS ASSIGNED TO THE EFFECTIVE STRAIN
C***********************************************************************
C
      DO 450 N=1,NUMEL
  450 TEPS(4,N)=0.0001
C
      FRNFCE(NUMNP)=0.003
      DPNSTP=0.1
      DPNHED=UZ(NUMNP)
      ESTAR=1.0
C
C***********************************************************************
C     IF THE COMPUTATION IS INTERRUPTED AFTER A NUMBER OF STEPS
C     AND RESTARTED, THEN NECESSARY DATA NEED BE FEED
C***********************************************************************
C
      IF(NREAD .LE. 0) GO TO 440
      READ(5,1017) (UR(I), UZ(I), SLOP(I), I=1,NUMNP)
      READ(5,1017) ( R(I),  Z(I), I=1, NUMNP)
      READ(5,1017)((TEPS(I,N),I=1,4), N=1, NUMEL)
      READ(5,1017)(THICK(N),N=1,NUMEL )
      READ(5,2223)PNHED,NTOUCH,TCHCOF,EFACT
      READ(5,233)(FRNFCE(N),N=1,NUM1)
      READ 233,ESTAR,DPNSTP,DPNHED
  440 CONTINUE
C
      NSTEP=0
      TOUCH2=TDIST
 2100 NSTEP=NSTEP+1
C
C***********************************************************************
C     DPNHED=ASSIGNED INCREMENT OF PUNCH HEAD TRAVEL
C     ESTAR=ADJUSTING FACTOR
C***********************************************************************
```

```
C
      PNHED=PNHED+DPNHED
      DO 445 I=1,NUMNP
      UR(I)=UR(I)*ESTAR
      UZ(I)=UZ(I)*ESTAR
  445 CONTINUE
C
      TOUCH3=TOUCH2
      DO 247 I=1,NCHECK
  247 TOUCH(I)=0.
C
 2101 CONTINUE
      IF(ICHCOF .EQ. 1.)GO TO 219
      NTOUCH=NTOUCH-1
      NTCP1=NTOUCH+1
      FRNFCE(NTOUCH)=FRNFCE(NTCP1)/3.
  219 CONTINUE
  446 CONTINUE
C
C*********************************************************************
C     UPDATING OF THE BOUNDARY CONDITION
C*********************************************************************
C
C
      DO 100 I=2,NUM1
      CODE(I)=0.0
      IF(I .GE. NTOUCH)CODE(I)=4.0
  100 CONTINUE
      CODE(NUMNP)=3.0
      NTCHM1=NTOUCH-1
C
C*********************************************************************
C     COMPUTE THE YIELD STRESS AND THE WORKHARDENING RATE
C*********************************************************************
C
      DO 220 N=1, NUMEL
      CALL HARD(TEPS(4,N),YY(N))
      CALL HARD2(TEPS(4,N),YX(N))
  220 CONTINUE
C
C
C
C
      WRITE(6,1007) NSTEP
C
C
  659 CONTINUE
      WRITE(6,1031)
C
C*********************************************************************
C     DETAIL OF THE PRESENT CONFIGURATION
C     SPHI=SINE OF ANGLE PHI
C     CPHI=COSINE OF ANGLE PHI
C     DL=ELEMENT LENGTH
C*********************************************************************
C
      DO 690 N=1, NUMEL
      NP1=N+1
      DR=R(N)-R(NP1)
      DZ=Z(NP1)-Z(N)
      DL(N)=SQRT(DR*DR+DZ*DZ)
      SPHI(N)=DR/DL(N)
      CPHI(N)=DZ/DL(N)
      PHI(N)=ASIN(SPHI(N))*180./3.14156
      WRITE(6,1030)N,PHI(N),THICK(N),DL(N)
  690 CONTINUE
C
C
      RP1=RVALUE+1.
      RCONST=3.*RP1/( 2.*(1.+RVALUE+RVALUE) )
      D(1,1)=RP1*RCONST
      D(1,2)=RVALUE*RCONST
      D(2,1)=D(1,2)
      D(2,2)=D(1,1)
      CLAMDA=RP1*RP1/(1.+RVALUE+RVALUE)
      POISON=RVALUE/RP1
C
C
      K=0
 2001 CONTINUE
      K=K+1
C
C
      CALL STIFF(R,Z,UR,UZ,CODE,SLOP,YY,YX,SPHI,CPHI,DL,EPS,
     1THICK,ALPHA,GAMMA,ETA,FRNFCE,FF,A,B,NG)
```

```
C
C***********************************************************************
C      INTRODUCTION OF BOUNDARY CONDITION
C***********************************************************************
C
C
       CALL MODIFY(CODE,A,B,ALPHA,GAMMA,ETA,NUMNE,NEQ,MBAND,ERNECE)
C
C      BANDED SYMMETRIC SOLUTION
C
       CALL TRIA(NEQ,MBAND,A)
       CALL BACKS(NEQ,MBAND,A,B)
C
C***********************************************************************
C      PERTURBATION OF UR IS COMPUTED FROM PERTURBATION OF UZ FOR NODES
C***********************************************************************
C
C
       DO 101 N=2,NUM1
       IR=3*N-2
       IZ=IR+1
       IF(N .GE. NTOLCH)B(IR)=B(IZ)/ALPHA(N)+GAMMA(N)
  101  CONTINUE
C
C***********************************************************************
C      TO OBTAIN AN EFFICIENT CONVERGENCE ACOEF IS COMPUTED
C      THE ACCELERATING COEFFICIENT IS DETERMINED IN SUCH A WAY
C      PERTURBATIONAL TERM TIMES ACCELERATING COEFFICIENT IS NEVER
C      GREATER THAN THE INITIAL VALUE, BUT BE A FRACTION OF IT.
C      E.G. A2=2 MEANS HALF
C***********************************************************************
C
       CONCOF=0.0
       DO 103 I=1,NUMNE
       UUR(I)=UR(I)
       UUZ(I)=UZ(I)
       IZ=3*I-1
       IR=IZ-1
       IF(B(IR) .EQ. 0.)GO TO 102
       COF1=UR(I)/B(IR)
  102  IF(B(IZ) .EQ. 0.)GO TO 103
       COF2=UZ(I)/B(IZ)
       COF1=ABS(COF1)
       COF2=ABS(COF2)
       A1=AMIN1(A1,COF1,COF2)
  103  CONTINUE
C
  105  CONTINUE
       IF(CONCOF .EQ. 1.)A2=A2*5.
       IF(A2 .GT. 1250)GO TO 2300
       IF(A1 .EQ. 1.0 .AND. A2 .GT. 10.)COF1=5.
       IF(COF1  .EQ. 0.)COF1=2.
       A1=COF1/A2
       IF(A1 .GT. 1.0) A1=1.0
       ACOEF=A1
       WRITE(6,104)A1
C
C***********************************************************************
C      OBTAIN NEW VALUE
C***********************************************************************
C
       DO 130 I=1, NUMNE
       IZ=3*I-1
       IR=IZ-1
       IS=IZ+1
       UR(I)=UUR(I)
       UZ(I)=UUZ(I)
       UR(I)=UR(I)+B(IR)*ACOEF
       UZ(I)=UZ(I)+B(IZ)*ACOEF
       SLOP(I)=SLOP(I)+B(IS)*ACOEF
  130  CONTINUE
C
       WRITE(6,1016) K
       WRITE(6,1006) K
C
C
C***********************************************************************
C      COMPUTE NORM OF ERROR AND NORM OF SOLUTION.
C***********************************************************************
C
       ENORM = 0.
       SNORM = 0.
       DO 134 I=1, NUMNP
       IZ=3*I-1
       IR=IZ-1
```

```
      IS=IZ+1
      ENORM = ENORM + B(IR)*B(IR) + B(IZ)*B(IZ) + B(IS)*B(IS)
      SNORM = SNORM + UR(I)*UR(I) + UZ(I)*UZ(I) + SLOP(I)*SLOP(I)
  134 CONTINUE
      ENORM = SQRT(ENORM)
      SNORM = SQRT(SNORM)
      ESNORM=ENORM/SNORM
      WRITE(6,1015) SNORM,ENORM,ESNORM
C
      DO 776 I=1, NUMNP
      IZ=3*I-1
      IR=IZ-1
      IS=IZ+1
      WRITE(6,1002) I,B(IR),B(IZ),B(IS),UR(I),UZ(I),SLOP(I),R(I),Z(I)
  776 CONTINUE
C
  131 CONTINUE
C
C
C
C
C***********************************************************************
C     COMPUTE STRAIN FROM THE NEW GUESS.
C     EPS(1,N)=INCREMENT OF MERIDIAN STRAIN
C     EPS(2,N)=INCREMENT OF TANGENTIAL STRAIN
C     EPS(3,N)=INCREMENT OF THICKNESS STRAIN
C***********************************************************************
C
C
      DO 800 N=1,NUMEL
      NP1=N+1
      DLL=DL(N)
      SPH=SPHI(N)
      CPH=CPHI(N)
      AU=UR(N)+UR(NP1)
      AR=R(N)+R(NP1)
      DR=R(N)-R(NP1)
      DZ=Z(NP1)-Z(N)
      DU=UR(N)-UR(NP1)
      DW=UZ(NP1)-UZ(N)
      EX1=1.+2.*DR*DU/DLL/DLL+2.*DZ*DW/DLL/DLL+(DU*DU+DW*DW)/DLL/DLL
      EPS(1,N)=SQRT(EX1)-1.
      EPS(1,N)=ALOG(1.+EPS(1,N))
      EPS(2,N)=AU/AR
      EPS(3,N)=-EPS(1,N)-EPS(2,N)
  800 CONTINUE
C
C
C
C
      TEST=0.0
      WRITE(6,1026) NSTEP
C
C
C
C***********************************************************************
C     COMPUTE INCREMENT OF EFFECTIVE STRAIN
C***********************************************************************
C
      DO 222 N=1, NUMEL
      ES=EPS(1,N)
      ET=EPS(2,N)
      RBAR=RP1*(ES*ES+ET*ET) + 2.*RVALUE*ES*ET
      EPS(4,N)=SQRT(2.*RCONST*RBAR/3.)
C
C
      IF(NSTEP .EQ. 0)YY(N)=YIELD
      STS(1,N)=CLAMDA*(ES + POISON*ET)*YY(N)/EPS(4,N)
      STS(2,N)=CLAMDA*(ET + POISON*ES)*YY(N)/EPS(4,N)
C
C***********************************************************************
C     COMPUTE STRESS DISTRIBUTION
C     STS(1,N)=MERIDIAN STRESS
C     STS(2,N)=CIRCUMFERENTIAL STRESS
C     STS(3,N)=EFFECTIVE STRESS
C***********************************************************************
C
      IF(ESNORM .LT. FLIMIT)TEST=1.0
      ES=STS(1,N)
      ET=STS(2,N)
      EFSTRS=ES*ES+ET*ET-2.*POISON*ES*ET
      STS(3,N)= SQRT(EFSTRS)
C
      WRITE(6,1003)N,(EPS(I,N),I=1,4)
  222 CONTINUE
```

```
C
      WRITE(6,1027)
      DO 430 N=1,NUMEL
  430 WRITE(6,2251)N,(STS(I,N),I=1,3)
C
C***********************************************************************
C     CHECK WHETHER ACOEF IS TOO LARGE TO CAUSE A PHYSICALLY
C     UNACCEPTABLE SOLUTION. WHENEVER COMPUTED MERIDIAN STRESS BECOMES
C     NEGATIVE ADJUST ACOEF VALUE)
C***********************************************************************
C
      CONCOF=0.0
      STS1=0.0
      DO 431 N=1,NUMEL
      IF(STS(2,N) .LT. STS1)STS1=STS(2,N)
  431 IF(STS(1,N) .LT. STS1)STS1=STS(1,N)
      IF(STS1 .LT. 0.)CONCOF=1.0
      IF(CONCOF .EQ. 1.0)GO TO 105
C
C
C
C***********************************************************************
C     CHECK WHETHER (ERROR NORM)/(SOLUTION NORM) IS LESS THAN FLIMIT
C     IF YES, THE SOLUTION IS FINAL
C***********************************************************************
C
C
      IF(ESNORM .LT. FLIMIT)GO TO 436
C
      IF(K .GE. ITER)GO TO 436
      GO TO 2001
 2000 CONTINUE
 2200 CONTINUE
C
  438 CONTINUE
      IF(ESNORM .GT. FLIMIT)GO TO 777
C
C
      WRITE(6,2800)
C
C     EXAMINATION ON BOUNDARY ASSUMPTIONS
C***********************************************************************
C
      DO 250 I=1,NCHECK
      N=NTOUCH-I
      IF(N .LE. 1)GO TO 250
      TOUCH(I)=(Z(N)+UZ(N)+PNRAD-PNHED)**2.+(R(N)+UR(N))**2.
      TOUCH(I)=SQRT(TOUCH(I))-PNRAD
      WRITE(6,2900)N,TOUCH(I)
  250 CONTINUE
C
      IF(ABS(TOUCH(1)) .LT. .0001)TOUCH(1)=0.
C
C***********************************************************************
C     CHECK ON BOUNDARY OVER PUNCH HEAD
C***********************************************************************
C
      IF(TOUCH(1) .GE. 0.)GO TO 3000
C
      WRITE(6,3100)
C
C***********************************************************************
C     NODE AT NCHM1 IS INSIDE PUNCH, COMPUTE AGAIN
C***********************************************************************
C
      TCHCOF=0.0
      TINSDE=0.0
      GO TO 2101
C
C
 3000 CONTINUE
      TINSDE=1.0
      TEST=1.0
C
C***********************************************************************
C     COMPUTATION OF FRICTION COEFFICIENT
C***********************************************************************
C
      MUDEX=0
      PPNHED=PNHED
      WRITE(5,233)(FRNFCE(N),N=NTOUCH,NUM1)
      IF(FRITN .EQ. 0.)GO TO 234
      WRITE(6,231)
C
      DO 230 I=NTOUCH,NUM1
```

```
      IZ=I*3-1
      IR=IZ-1
      DUM1=(Z(I)+UZ(I)+PNEAD-DPNHED)/PNEAD
      DUM2=(R(I)+UR(I))/PNEAD
      PN=FF(IZ)*DUM1+FF(IR)*DUM2
      PT=FF(IZ)*DUM2-FF(IR)*DUM1
      XMU=PT/PN
      WRITE(6,232)I,XMU
      XMU=XMU/FRITN
      IF(XMU .GT. 1.02 .OR. XMU .LT. .98)MUDEX=1
      FRNFCE(I)=FRNFCE(I)/XMU
  230 CONTINUE
  234 CONTINUE
C
C
C
C
C***********************************************************************
C     MUDEX=0, IF FRICTION CONDITION IS SATISFIED
C          =1, OTHERWISE
C***********************************************************************
C
C
      IF(MUDEX .EQ. 1)TCHCOF=1.0
      IF(MUDEX .EQ. 1)GO TO 2001
      NZZ=3*NTOUCH-1
C
C
C***********************************************************************
C     GENERALIZED NODAL FORCE NORMAL TO THE PUNCH IS COMPUTED
C     TO CHECK WHETHER THE BOUNDARY IS ASSUMED TO MOVE TOO FAST.
C***********************************************************************
C
      IF(FF(NZZ) .GT. 0.) GO TO 500
      IF(ABS(FF(NZZ)) .LT. .000001 .OR. TINSDE .EQ. 0.)GO TO 500
      TDIST=TOUCH3*.8
      WRITE(6,510)TDIST
      NTOUCH=NTOUCH+1
      TCHCOF=1.0
      GO TO 446
  500 CONTINUE
C
C
C
      NNTCH=NTOUCH
      DO 240 I=1,NCHECK
      IF(TOUCH(I)-TDIST)245,246,246
  245 NTOUCH=NNTCH-I
      TOUCH2=TOUCH(I)
      FRNFCE(NTOUCH)=FRNFCE(NNTCH)
  240 CONTINUE
  246 TCHCOF=1.0
      WRITE(6,1043)
C
C***********************************************************************
C     COMPUTE TOTAL STRAIN
C***********************************************************************
C
      DO 444 N=1,NUMEL
      DO 443 I=1, 4
  443 TEPS(I,N)=TEPS(I,N)+EPS(I,N)*TEST
      IF(ESNORM .LT. FLIMIT)THICK(N)=THICK(N)*EXP(EPS(3,N)*TEST)
      WRITE(6,1003)N,(TEPS(I,N),I=1,4)
  444 CONTINUE
C
C
C
C
C
C***********************************************************************
C     COMPUTE DPNHED FOR NEXT STEP WHICH WOULD GIVE THE INCREMENT
C     OF MAXIMUM EFFECTIVE STRAIN APPROXIMATELY EQUAL TO PRESET VALUE.
C***********************************************************************
C
      EMAX=0.0
      DO 775 N=1,NUMEL
  775 IF(EPS(4,N) .GT. EMAX) EMAX=EPS(4,N)
      EFACTA=ECONST/EMAX
      IF(NSTEP .LE. 2  .AND.  ITCONT .EQ. 0)GO TO 778
      DPNHED=2./EFACTA/UZ(NUMNP)-1./EFACT/DPNSTR
      DPNHED=1./DPNHED
      ESTAR=DPNHED/UZ(NUMNP)
      DPNSTR=UZ(NUMNP)
  778 CONTINUE
      EFACT=EFACTA
```

```
C
C
C
  777 CONTINUE
      IF(ESNORM .GT. FLIMIT)TCHCOF=1.0
C
C*****************************************************************
C     NEW CONFIGURATION
C*****************************************************************
C
      DO 439 I=1, NUMNP
      IZ=3*I-1
      IR=IZ-1
      R(I)=R(I)+UR(I)*TEST
      Z(I)=Z(I)+UZ(I)*TEST
  439 CONTINUE
C
C
C*****************************************************************
C     PUNCH THE SOLUTION
C*****************************************************************
C
C
      IF(NPUNCH .EQ. 0) GO TO 310
      PUNCH 1017, (UR(I),UZ(I), SLOP(I), I=1,NUMNP)
      PUNCH 1017, (R(I),Z(I),I=1, NUMNP)
      PUNCH 1017, ( (TEPS(I,N),I=1,4), N=1, NUMEL)
      PUNCH 1017,(THICK(N),N=1,NUMEL)
      PUNCH 2223,PNHED,NTOUCH,TCHCOF,EFACT
      PUNCH 233,(FRNFCE(N),N=1,NUM1)
      PUNCH 233,ESTAR,DENSTR,DPNHED
  310 CONTINUE
C
      IF(ESNORM .GT. FLIMIT)GO TO 2300
C
      WRITE(6,1040)
      DO 849 I=1,NUMNP
      IR=3*I-2
      IZ=IR+1
      IL=IZ+1
      WRITE(6,1041)(I,FF(IR),FF(IZ),FF(IL))
  849 CONTINUE
C
C
C*****************************************************************
C     COMPUTE THE PUNCH LOAD FROM ENERGY BALANCE
C*****************************************************************
C
      SUMF=0.
      DO 850 I=2,NUMNP
      IZ=I*3-1
      IR=IZ-1
  850 SUMF=SUMF+FF(IZ)*UZ(I)+FF(IR)*UR(I)
      SUMF=SUMF/UZ(NUMNP)/TEST/RADIUS
      WRITE(6,1042)SUMF
      IF(NSTOP .EQ. 1) GO TO 2300
      WRITE(6,1028)PPNHED,NNTCH
      WRITE(6,1029)EMAX,UZ(NUMNP),EFACT
      IF(NSTEP .LT. NECEM) GO TO 2100
 2300 CONTINUE
 2301 CONTINUE
C
C
 1002 FORMAT(I5,3F13.7,5X,3F13.7,5X,3F13.7)
 1003 FORMAT(I7,11F11.6)
 1004 FORMAT(16I5)
 1005 FORMAT(1H1,* STRAIN-STRESS SOLUTION AT  STEP  NUMBER  =*,I4//
     1 * EL. NO...R-STRAIN...Z-STRAIN..TH-STRAIN...RZ-STRAIN...EF-STRAIN
     2...R-STRES...Z-STRES...TH-STRES...RZ-STRES..EF-STRES..AVG-STRES..*
     3)
 1006 FORMAT(///  30X, * DISPLACEMENT SOLUTION AT ITERATION NUMBER =*,I4
     1///  20X, * PERTURBED*, 26X, * TOTAL*, 20X, * DEFORMED COORD*/
     2/   *   NP          DU            DW              DBETA              U
     3        W                   BETA               R              Z*)
 1007 FORMAT(1H1,70X,*ITERATION PROCESS FOR STEP*,I4)
 1008 FORMAT(       60X, * TOTAL R-LOAD =*, F12.7
     1           /  60X, * TOTAL Z-LOAD =*, F12.7
     2           /  60X, * TOTAL M-LOAD =*, F12.7)
 1010 FORMAT(    // * NODAL POINT FORCE AT STEP =*, I4//
     1*.......N.P.........R-FORCE.......Z-FORCE......Z-STRESS ON DIE SUR
     2FACE...*)
 1011 FORMAT(I5,3F10.0)
 1012 FORMAT(I5,6F17.5)
 1015 FORMAT(60X,* VELOCITY CONVERGENCE*,/
     1                 60X, * NORM OF SOLUTION VECTOR =*, F13.8
```

```
     1           /  50X, * NORM OF   ERROR  VECTOR =*, F13.8
     2           /  60X, * FRACTIONAL' NORM        =*, F13.8)
 1016 FORMAT(          * DISPLACEMENT SOLUTION AT ITERATION NUMBER =*,I4)
 1017 FORMAT(8F10.7)
 1018 FORMAT(////* DOES NOT CONVERGE*//
     1* TRY AGAIN WITH DECELLERATION COEFFICIENT =ACOEF= LESS THAN*,
     2F8.3)
 1020 FORMAT(20F4.1)
 1025 FORMAT(4X,I5,3X,F12.6,10X,I5,3X,F12.6,10X,I5,3X,F12.6)
 2251 FORMAT(I5,4F20.7)
 1026 FORMAT(///*INCREMENTAL STRAIN-TOTAL STRAIN AT STEP NUMBER=*,I4//
     1*EL  NO....S-STRAIN......THE-STRAIN.......THI-STRAIN......EF-STRAN
     1*)
 1027 FORMAT(///*EL.  NO....S-STRESS....THE-STRESS....EF-STRESS....*)
 1042 FORMAT(*   PUNCH  FORCE=*,F15.7)
 1043 FORMAT(///*EL  NO....S-STRAIN......THE-STRAIN......THI-STRAIN....
     1EF-STRAIN...*)
 1041 FORMAT(5X,I10,5X,3F20.7)
 1040 FORMAT(//*     NO.  OF  NODE                  FORCE*)
  510 FORMAT(//*   NTOUCH IS FORCED TO TOUCH, COMPUTE  AGAIN
     1*,/*   TDIST=*,F10.7)
 1030 FORMAT(/I7,3F10.5)
 1031 FORMAT(*  GEOMETRY  OF  PROFILE*//
     1 *EL  NNO.....ANGLE.....THICKNESS.....*)
  104 FORMAT(//*      ACOEF  CALCULATED*,F10.7)
 2900 FORMAT(/I10,F20.7)
 2800 FORMAT(//*  CHECKING  DISTANCE  AWAY  FROM  PUNCH*,/
     1*  ELEM  NO.           TOUCH*/)
 3100 FORMAT(*  NODE  AT  NCHM1 IS INSIDE PUNCH, COMP  AGAIN*)
  231 FORMAT(//*   NODAL  POINT            COEFFICIENT*,/)
  232 FORMAT(I10,F10.5)
  233 FORMAT(4F15.7)
 2223 FORMAT(F15.7,I5,3F15.7)
 1028 FORMAT(*   PUNCH  HEAD  DISPLACEMENT*,F10.5/*    NTOUCH=  *,I5)
 1029 FORMAT(///*   MAX  EFFECTIVE  STRAIN  INCREMENT=*,F10.7/*  PUNCH
     1 HEAD INCREMENT=*,F10.7,/*  PUNCH  HEAD  ADJUSTING  FACTOR=*,F10.7
     2)
C
      RETURN
      END
```

```
      SUBROUTINE STIFF(R,Z,UR,UZ,CODE,SLOP,YY,YX,SPHI,CPHI,DL,EPS,
     1THICK,ALPHA,GAMMA,ETA,ERNFCE,FF,A,B,NQ)
C
      COMMON/GENCON/NUMNP,NUMEL,HED(12),DLL,NEQ,NFORM,YIELD,TEST,ITER,
     1NREAD,NPUNCH,NPRINT,PVALUE,T,MBAND,ENRAD,RADIUS,FRITN,
     2ECCNST,ENHED,TDIST
      COMMON/STFMAT/H(6),P(6,6),TEX,TEY,TEZ,THKL
      COMMON/CONQUAD/SS(4),WT(4),D(2,2),SQRT1
      COMMON/ATOUCH/NTOUCH
C
      DIMENSION R(1),Z(1),CODE(1),UR(1),UZ(1),SLOP(1),B(1),A(NQ,1),
     1 EPS(4,1),RR(2),ZZ(2),UU(6),YY(1),THICK(1),DL(1),SPHI(1),CPHI(1),
     2 YX(1),ALPHA(1),GAMMA(1),ETA(1),ERNFCE(1),FF(1)
C
      DO 50 N=1, NEQ
      B(N)=0.
      DO 50 M=1,MBAND
   50 A(N,M)=0.
C
      WT(1)=0.3478548451
      WT(2)=0.6521451549
      WT(3)=WT(1)
      WT(4)=WT(2)
C
C
      DO 1000 N=1, NUMEL
      NP1=N+1
      DLL=DL(N)
      SPH=SPHI(N)
      CPH=CPHI(N)
      RR(1)=R(N)
      ZZ(1)=Z(N)
      RR(2)=R(NP1)
      UU(1)=UR(N)
      UU(2)=UZ(N)
      UU(3)=SLOP(N)
      UU(4)=UR(NP1)
      UU(5)=UZ(NP1)
      UU(6)=SLOP(NP1)
      THKL=THICK(N)*DLL
```

```
      ZZ(2)=Z(NP1)
      YG=YX(N)
      YH=YY(N)
      DZ=Z(NP1)-Z(N)
C
      CALL QUAD(RR,ZZ,UU,DLL,SPH,CPH,YG,YH)
C
C
C
C***********************************************************************
C     PERFORM THE ASSEMBLY OPERATION. BECAUSE MATRIX A IS SYMMETRIC
C     ONLY UPPER HALF OF THE MATRIX IS CREATED.  AND THE STORAGE FOR
C     MATRIX A IS A SQUARE ARRAY BECAUSE OF BANDED SYMMETRIC PROPERTY
C***********************************************************************
C
      DO 200 I=1, 6
      II=N*3 - 3 + I
      B(II)=B(II)+H(I)
      DO 200 J=1, 6
      JJ=N*3-3+J-II+1
      IF(JJ .LT. 1) GO TO 200
      A(II,JJ)=A(II,JJ)+P(I,J)
  200 CONTINUE
C
 1000 CONTINUE
C
C
C
C***********************************************************************
C     TO HANDLE MIXED BOUNDARY CONDITION, FOLLOWING MATRICES ARE EVALUATED
C***********************************************************************
C
      C=PNRAD-PNHED
      NMP1=NUMNP-1
C
      DO 1200 N=2,NMP1
      DUM2=R(N)+UR(N)
      DUM1=C+Z(N)+UZ(N)
      NP1=N+1
      NM1=N-1
C     DUM3=(2.*R(N)+R(NP1))/3.*DL(N)+(2.*R(N)+R(NM1))/3.*DL(NM1)     /2
C     IF(N .EQ. NTOUCH)DUM3=(2.*R(N)+R(NP1))/3.*DL(N)+(3.*R(N)+R(NM1))/
C    16.*DL(NM1)
      ALPHA(N)=-DUM2/DUM1
      GAMMA(N)=(PNRAD*PNRAD-DUM2*DUM2-DUM1*DUM1)/2./DUM2
      ETA(N)=FPNFCE(N)*PNRAD/DUM2
 1200 CONTINUE
C
C
C***********************************************************************
C     STORE GENERALIZED NODAL FORCE
C***********************************************************************
C
      NUM3=NUMNP*3
      DO 1300 I=1,NUM3
 1300 FF(I)=-B(I)
C
C
 1001 FORMAT(///,* THE DIAGONAL VECTOR OF MATRIX OF STIFFNESS*/)
 1002 FORMAT(12E11.3)
 1005 FORMAT(// 29H ELEMENT WITH NEGATIVE AREA =, I5)
C
      RETURN
      END
```

```
      SUBROUTINE QUAD(RR,ZZ,UU,DLL,SPH,CPH,S2,S1)
C
      COMMON/STFMAT/H(6),P(6,6),TEX,TEY,TEZ,THKL
      COMMON/CONQUAD/SS(4),WT(4),D(2,2),SQRT1
      COMMON/ISO/RRVAL
C
      DIMENSION RR(2),ZZ(2),UU(6),B(2,6),XX(6,6),BZERO(6),DB(2,6)
C
      RVALUE=RRVAL
      RC=(RR(1)+RR(2))/2.
C
      DO 2 I=1,6
      H(I)=0.
      DO 2 K=1, 2
      DO 2 J=1,6
    2 P(I,J)=0.
```

```
C
C
      DZ=ZZ(2)-ZZ(1)
      DR=RR(1)-RR(2)
      DU=UU(1)-UU(4)
      DW=UU(5)-UU(2)
      AU=UU(1)+UU(4)
      AR=RR(1)+RR(2)
C
      C1=2.*DR/DLL/DLL
      C2=2.*DU/DLL/DLL
      C3=2.*DZ/DLL/DLL
      C4=2.*DW/DLL/DLL
      C5=AU/AR/2.
      C6=1.+DR*C2+DZ*C4+(DU*DU+DW*DW)/DLL/DLL
      C7=2./DLL/DLL
      C8=2./AR/AR
      C9=1./SQRT(C6)/2.
      C10=C9/C6
      C11=C1+C2
      C12=C3+C4
C
C
      DES1=SQRT(C6)
      DET1=2.*C5+1.
C
      E1=C9*C11/DES1
      E2=-C9*C12/DES1
      E3=-E1
      E4=-E2
      E5=1./AR/DET1
      E6=(-C10*C11*C11/2.+C9*C7)/DES1-E1*E1
      E7=-E6
      E8=C10*C11*C12/2./DES1-E1*E2
      E9=-E8
      E10=E8
      E11=(-C10*C12*C12/2.+C9*C7)/DES1-E2*E2
      E12=-E5*E5
C
C
C
C
C***********************************************************************
C     COMPUTATION OF EFFECTIVE STRAIN INCREMENT
C     DES=MERIDIAN STRAIN INCREMENT
C     DET=CIRCUMFERENTIAL STRAIN INCREMENT
C     E1=DERIVATIVE OF MERIDIAN STRAIN INCREMENT WITH RESPECT TO UU(1)
C        =D(DES)/D(UU(1))
C     E2=D(DES)/D(UU(2))
C     E3=D(DES)/D(UU(4))
C     E4=D(DES)/D(UU(5))
C     E5=D(DET)/D(UU(1))
C     E6=D(E1)/D(UU(1))
C     E7=D(E1)/D(UU(4))
C     E8=D(E1)/D(UU(2))
C     E9=D(E3)/D(UU(2))
C     E10=D(E4)/D(UU(5))
C     E11=D(E5)/D(UU(2))
C***********************************************************************
C
C
      DES=ALOG(DES1)
      DET=ALOG(DET1)
      RVP1=RVALUE+1.
      RVP2=SQRT(2.*RVALUE+1.)
      RVP3=RVP1/RVP2
      RVP4=2.*RVALUE/RVP1
C
C*********************************************************************
C     EFFECTIVE  STRAIN
C     F1=DERIVATIVE OF EFFECTIVE STRAIN INCREMENT WITH RESPECT TO UU(1)
C     F2=WITH RESPECT TO UU(2)
C     F3=WITH RESPECT TO UU(4)
C     F4=WITH RESPECT TO UU(5)
C     F11=D(F1)/D(UU(1))
C     F12=D(F1)/D(UU(2))
C     F13=D(F1)/D(UU(4))
C     F14=D(F1)/D(UU(5))
C     F22=D(F2)/D(UU(2))
C     F23=D(F2)/D(UU(4))
C     F24=D(F2)/D(UU(5))
C     F33=D(F3)/D(UU(4))
C     F34=D(F3)/D(UU(5))
C     F44=D(F4)/D(UU(5))
C*********************************************************************
```

```
C
      EFS=DES*DES+DET*DET+RVP4*DES*DET
      EFS1=RVP3*SQRT(EFS)
      EFS2=RVP3/SQRT(EFS)/2.
      EFS3=-RVP3/EFS/SQRT(EFS)/4.
C
C
      D1=(2.*DES+RVP4*DET)*E1+(2.*DET+RVP4*DES)*E5
      D2=(2.*DES+RVP4*DET)*E2
      D3=(2.*DES+RVP4*DET)*E3+(2.*DET+RVP4*DES)*E5
      D4=(2.*DES+RVP4*DET)*E4
C
      F1=EFS2*D1
      F2=EFS2*D2
      F3=EFS2*D3
      F4=EFS2*D4
C
      F11=EFS3*D1*D1+EFS2*((2.*DES+RVP4*DET)*E6+(2.*DET+RVP4*DES)*E12
     1+(2.*E1+RVP4*E5)*E1+(2.*E5+RVP4*E1)*E5)
      F12=EFS3*D1*D2+EFS2*((2.*DES+RVP4*DET)*E8+(2.*E1+RVP4*E5)*E2)
      F13=EFS3*D1*D3+EFS2*((2.*DES+RVP4*DET)*E7+(2.*E3+RVP4*E5)*E1
     1+(2.*E5+RVP4*E3)*E5+(2.*DET+RVP4*DES)*E12)
      F14=EFS3*D1*D4+EFS2*((2.*DES+RVP4*DET)*E9+(2.*E1+RVP4*E5)*E4)
      F22=EFS3*D2*D2+EFS2*((2.*DES+RVP4*DET)*E11+2.*E2*E2)
      F23=EFS3*D2*D3+EFS2*((2.*DES+RVP4*DET)*E9+(2.*E3+RVP4*E5)*E2)
      F24=EFS3*D2*D4+EFS2*(-(2.*DES+RVP4*DET)*E11+2.*E4*E2)
      F33=EFS3*D3*D3+EFS2*((2.*DES+RVP4*DET)*E6+(2.*E3+RVP4*E5)*E3+(
     12.*E5+RVP4*E3)*E5+(2.*DET+RVP4*DES)*E12)
      F34=EFS3*D3*D4+EFS2*((2.*DES+RVP4*DET)*E10+(2.*E3+RVP4*E5)*E4)
      F44=EFS3*D4*D4+EFS2*((2.*DES+RVP4*DET)*E11+2.*E4*E4)
C
      P(1,1)=((S1+S2*EFS1)*F11+S2*F1*F1)*RC*THKL
      P(1,2)=((S1+S2*EFS1)*F12+S2*F2*F1)*RC*THKL
      P(1,4)=((S1+S2*EFS1)*F13+S2*F1*F3)*RC*THKL
      P(1,5)=((S1+S2*EFS1)*F14+S2*F1*F4)*RC*THKL
      P(2,2)=((S1+S2*EFS1)*F22+S2*F2*F2)*RC*THKL
      P(2,4)=((S1+S2*EFS1)*F23+S2*F2*F3)*RC*THKL
      P(2,5)=((S1+S2*EFS1)*F24+S2*F2*F4)*RC*THKL
      P(4,4)=((S1+S2*EFS1)*F33+S2*F3*F3)*RC*THKL
      P(4,5)=((S1+S2*EFS1)*F34+S2*F3*F4)*RC*THKL
      P(5,5)=((S1+S2*EFS1)*F44+S2*F4*F4)*RC*THKL
      P(2,1)=P(1,2)
      P(4,1)=P(1,4)
      P(4,2)=P(2,4)
      P(5,1)=P(1,5)
      P(5,2)=P(2,5)
      P(5,4)=P(4,5)
C
      H(1)=-(S1+S2*EFS1)*F1*RC*THKL
      H(2)=-(S1+S2*EFS1)*F2*RC*THKL
      H(4)=-(S1+S2*EFS1)*F3*RC*THKL
      H(5)=-(S1+S2*EFS1)*F4*RC*THKL
C
C
   71 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE CONDEN(A,B,NEQ,MBAND,N,U)
C
C*****************************************************************************
C     PERFORM THE MATRIX CONDENSATION WHEN THE VALUE OF A COMPONENT
C     X IN AX=B IS SPECIFIED EQUAL TO ZERO
C*****************************************************************************
C
      DIMENSION B(NEQ),A(NEQ,1)
C
      DO 250 M=2,MBAND
      K=N-M+1
      IF(K) 235,235,230
  230 B(K)=B(K)-A(K,M)*U
      A(K,M)=0.0
  235 K=N+M-1
      IF(NEQ-K) 250,240,240
  240 B(K)=B(K)-A(N,M)*U
      A(N,M)=0.0
  250 CONTINUE
      A(N,1)=1.0
      B(N)=U
      RETURN
C
      END
```

```
      SUBROUTINE MODIFY(CODE,A,B,ALPHA,GAMMA,ETA,NUMNE,NEQ,MBAND,FRNFCE)
C
      DIMENSION CODE(1),A(NEQ,1),B(1),ALPHA(1),GAMMA(1),ETA(1),FRNFCE(1)
C
      DO 121 I=1, NUMNE
      IL=3*I
      IZ=IL-1
      IR=IZ-1
      C=CODE(I)
      IF (C.EQ. 1.) GO TO 101
      IF (C.EQ. 2.) GO TO 102
      IF (C.EQ. 3.) GO TO 103
      CALL CONDEN(A,B,NEQ,MBAND,IL,C.)
      CONSTA=ALPHA(I)
      CONSTE=ETA(I)
      CONSTG=GAMMA(I)
      IF(C .EQ. 4)GO TO 104
      GO TO 121
C
  101 CONTINUE
      CALL CONDEN(A,B,NEQ,MBAND,IR,C.)
      CALL CONDEN(A,B,NEQ,MBAND,IL,0.)
      GO TO 121
C
  102 CONTINUE
      CALL CONDEN(A,B,NEQ,MBAND,IZ,C.)
      CALL CONDEN(A,B,NEQ,MBAND,IL,0.)
      GO TO 121
C
  103 CONTINUE
      CALL CONDEN(A,B,NEQ,MBAND,IR,0.)
      CALL CONDEN(A,B,NEQ,MBAND,IZ,0.)
      CALL CONDEN(A,B,NEQ,MBAND,IL,0.)
      GO TO 121
C
  104 CONTINUE
      CALL BCMIX(A,B,NEQ,MBAND,I,CONSTA,CONSTB,CONSTE,CONSTG)
  121 CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE TRIA(NN,MM,A)
      DIMENSION A(NN,1)
C
C**************************************************************************
C     TRIANGULIZATION OF GAUSSIAN ELIMINATION FOR THE SOLUTION
C     OF BANDED SYMMETRIC MATRIX
C**************************************************************************
C
      N=0
  100 N=N+1
      IF(N.EQ.NN)RETURN
      IF(A(N,1).NE.C.) GO TO 150
      GO TO 100
C
  150 I=N
      MB=MIN0(MM,NN-N+1)
C
      DO 260 L=2,MB
      I=I+1
      C=A(N,L)/A(N,1)
      IF(C.EQ.0.0)GO TO 260
      J=0
      DO 250 K=L,MB
      J=J+1
  250 A(I,J)=A(I,J)-C*A(N,K)
      A(N,L)=C
  260 CONTINUE
C
      GO TO 100
C
C
      END
```

```
      SUBROUTINE BACKS(NN,MM,A,B)
C
C
C**********************************************************************
C      BACK SUBSTITUTION FOR SOLUTION OF BANDED SYMMETRIC MATRIX
C**********************************************************************
C
C
      DIMENSION A(1),B(1)
C
      MMM=MM-1
      N=0
  270 N=N+1
      C=B(N)
      IF(A(N).NE.0.0)B(N)=B(N)/A(N)
      IF(N.EQ.NN)GO TO 300
      IL=N+1
      IH=MINO(NN,N+MMM)
      M=N
C
      DO 285 I=IL,IH
      M=M+NN
  285 B(I)=B(I)-A(M)*C
C
      GO TO 270
C
  300 IL=N
      N=N-1
      IF(N.EQ.0) RETURN
      IH=MINO(NN,N+MMM)
      M=N
C
      DO 400 I=IL,IH
      M=M+NN
  400 B(N)=B(N)-A(M)*B(I)
C
      GO TO 300
C
      END
```

```
      SUBROUTINE BCMIX(A,B,NEQ,MBAND,N,CONSTA,CONSTB,CONSTE,CONSTG)
C
C**********************************************************************
C      THIS SUBROUTINE HANDLES THE MIXED BOUNDARY CONDITION
C**********************************************************************
C
      DIMENSION A(NEQ,1),B(1)
      IR=3*N-2
      IZ=IR+1
      A(IZ,1)=A(IZ,1)+2.*A(IR,2)/CONSTA+A(IR,1)/CONSTA/CONSTA
      B(IZ)=B(IZ)+B(IR)/CONSTA-CONSTG*(A(IR,1)/CONSTA+A(IR,2))
      B(IZ)=B(IZ)+CONSTE
      A(IR,1)=1.0
      A(IR,2)=0.0
      B(IR)=0.0
C
      DO 100 M=3,MBAND
      K=IZ-M+1
      IF(K .LT. 1)GO TO 100
      MM=M-1
      A(K,M)=A(K,M)+A(K,MM)/CONSTA
      B(K)=B(K)-A(K,MM)*CONSTG
  100 A(K,MM)=0.0
C
      MB1=MBAND-1
C
      DO 200 M=2,MB1
      MM=M+1
      A(IZ,M)=A(IZ,M)+A(IR,MM)/CONSTA
      IZZ=IR+M
      B(IZZ)=B(IZZ)-A(IR,MM)*CONSTG
  200 A(IR,MM)=0.0
C
      RETURN
      END
```

```
      SUBROUTINE HARD(EPS,Y)
```

```
C
C*****************************************************************************
C      WORKHARDENING CHARACTERISTIC CURVE
C*****************************************************************************
C
       COMMON/MATERL/YVALUE,PRESTN,EXPNT,PRESTS
C
       Y=YVALUE*(PRESTN+EPS)**EXPNT+PRESTS
C
       RETURN
       END


       SUBROUTINE HARD2(EPS,Y)
C
C*****************************************************************************
C      COMPUTE WORK HARDENING RATE
C*****************************************************************************
C
       COMMON/MATERL/YVALUE,PRESTN,EXPNT,PRESTS
C
       Y=EXPNT*YVALUE*(PRESTN+EPS)**(EXPNT-1.)
C
       RETURN
       END
```

APPENDIX D

PROGRAM FOR THE ANALYSIS OF DEEP DRAWING AND PUNCH STRETCHING WITH ROUND DIE CORNER

This program is for the analysis of deep drawing with a hemispherical punch head and stretching with a hemispherical punch head. In stretching, a round die profile is considered.

(I) Data preparation card

1. Read HED (A 12)
2. Read RVALUE, T, ACOEF (5F 10.0)
3. Read ITER, NREAD, ITCONT, NFORM, NPUNCH, NPRINT, FLIMIT (6I5, F10.0)
4. Read NUMNP, NDEX (6I5)

   NDEX: 2, if punch stretching is to be analyzed
   3, if deep drawing is to be analyzed
5. Read PNRAD, RADIUS, DIERAD, RTART (4F 10.0)

   DIERAD: Radius of the die profile

   RTHRT: Distance from the pole to the die throat
6. Read FRITNP, FRITND, BHFCE (4F 10.0)

   FRITNP: Friction coefficient between the punch head and the blank

   FRITND: Friction coefficient between the die and the blank

   BHFCE: Blank holding force
   Set 0.0 for punch stretching problem
7. Read YVALUE, PRESTN, EXPNT, PRESTS (4F 10.0)
8. Read TCONTC, TDIST, ECONST (4F 10.0)

   TCONTC: Criterion distance of the contact with the die profile
   To start with, set this 0.002
9. Read N, CODE(N), R(N), Z(N), UR(N), UZ(N), SLOP(N), (I5, F5.0, 5F 10.0)
10. If NREAD = 1, the new input data is to be placed behind the nodal information card.

```
      PROGRAM SHEET(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,PUNCH)
      COMMON/GENCON/NUMNP,NUMEL,HED(12),DLL,NEQ,NFORM,YIELD,TEST,ITER,
     1NREAD,NPUNCH,NPRINT,RVALUE,T,MBAND,PNRAD,RADIUS,FRITNP,FRITND,
     2ECONST,FNHED,RTHRT,DIERAD,ICONTC,TDIST,BHFCE
C
C***********************************************************************
C     PROGRAM BOTH FOR PUNCH STRETCHING WITH ROUND PROFILE AND
C     FOR DEEP DRAWING, BY J.H.KIM
C***********************************************************************
C
      COMMON/MATERL/YVALUE,PRESTN,EXPNT,PRESTS
      COMMON/ISOTPY/RVAL1
C
C
C***********************************************************************
C     PROGRAM SHEET IS FOR CONTROLLING THE DIMENSION OF THE COMPLETE
C     PROGRAM.  ITS PURPOSE IS TO PREVENT ASSIGNING A LARGER THAN
C     NECESSARY DIMENSION FOR ANY ARRAY THROUGH THE USE OF THE
C     FOLLOWING STATEMENT
C***********************************************************************
C
      COMMON A(5000)
C
C
C***********************************************************************
C     NFIELD IS THE DIMENSION OF ARRAY A.  ITS VALUE CAN BE DETERMINED
C     PRECISELY BY RUNNING THE PROGRAM ONCE.
C***********************************************************************
C
      NFIELD=5000
C
C
C***********************************************************************
C     READ THE INPUT DATA CONTROL CARDS
C***********************************************************************
C
      TEST=1.
      READ(5,1000) HED
      READ(5,1004) RVALUE,T,ACOEF
      READ(5,1003) ITER,NREAD,ITCONT,NFORM,NPUNCH,NPRINT,FLIMIT
      READ(5,1003) NUMNP,NDEX
      READ(5,1004) PNRAD,RADIUS,DIERAD,RTHRT
      READ(5,1004) FRITNP,FRITND,BHFCE
      READ(5,1004) YVALUE,PRESTN,EXPNT,PRESTS
      READ(5,1004) ICONTC,TDIST,ECONST
C
C***********************************************************************
C     HED=OUTPUT TITLE
C     RVALUE=VALUE OF THE ANISOTROPY PARAMETER
C     ACOEF=ACCELERATING OR DECELERATING COEFFICIENT OF CONVERGENCE
C     NREAD=0, IF TO BYPASS THE READING STATEMENT IN SUBROUTINE PLAST
C     ITCONT=0, IF COMPUTATION STARTS AT THE VERY BEGGINNING AND FIRST/
C              SECOND STEPS ARE INCLUDED IN THE STEPS TO BE COMPUTED
C           =1, OTHERWISE
C            THIS INDEX IS RELATED TO THE DETERMINATION OF STEP SIZE
C     NFORM=NUMBER OF STEPS ASSIGNED PER RUN
C     NPUNCH=1, IF DATA ARE TO BE PUNCHED
C           =0, OTHERWISE
C     FLIMIT=VALUE OF (ERROR NORM)/(SOLUTION NORM) REQUIRED
C     FOR CONVERGENCE
C     NPRINT=1, IF NODAL POINT DATA ARE TO BE PRINTED
C           =0, OTHERWISE
C     NUMNP=NUMBER OF NODAL POINTS
C     PNRAD=RADIUS OF HEMISPHERICAL PUNCH HEAD
C     RADIUS=RADIUS OF THE BLANK
C     FRITNP=FRICTION COEFFICIENT BETWEEN BLANK AND PUNCH
C     FRITND=FRICTION BETWEEN BLANK AND DIE PROFILE
C     DIERAD=RADIUS OF DIE PROFILE
C     RTHRT=RADIUS OF DIE THROAT
C     ECONST=STEP SIZE IN MAXIMUM EFFECTIVE STRAIN INCREMENT
C     NDEX=2, IF PUNCH STRETCHING WITH ROUND PROFILE
C         =3, IF DEEP DRAWING
C
C
C     YVALUE, PRESTN, EXPNT, PRESTS ARE TO EXPRESS THE WORKHARDENING
C     CHARACTERISTICS OF THE BLANK
C     STRESS=YVALUE*(PRESTN+STRAIN)**EXPNT+PRESTS
```

```
C
C
C       NEQ=NUMBER OF EQUATIONS TO BE SOLVED
C       NUMEL=NUMBER OF ELEMENTS
C       MBAND=BAND WIDTH
C
C***********************************************************************
C
        RVAL1=RVALUE
        NUMEL=NUMNP-1
        MBAND=9
        NEQ=NUMNP*3
        NQ=NEQ
        NEL=NUMEL
C
C***********************************************************************
C       DETERMINE THE LOCATION OF THE STARTING POINTS OF DIFFERENT
C       ARRAYS ON ARRAY A
C***********************************************************************
C
C
        N1=1
        N2=N1+NUMNP
        N3=N2+NUMNP
        N4=N3+NUMNP
        N5=N4+NUMNP
        N6=N5+NUMNP
        N7=N6+NUMNP
        N8=N7+NUMEL
        N9=N8+NUMEL
        N10=N9+NUMEL
        N11=N10+NUMEL
        N12=N11+NUMEL
        N13=N12+NUMEL*3
        N14=N13+NUMEL*4
        N15=N14+NUMEL*4
        N16=N15+NEQ
        N17=N16+NEQ*MBAND
        N18=N17+NUMEL
        N19=N18+NUMNP
        N20=N19+NUMNP
        N21=N20+NUMNP
        N22=N21+NUMNP
        N23=N22+NUMNP
        N24=N23+NEQ
        N25=N24+NUMNP
        N26=N25+NUMNP
        N27=N26+NUMNP
        N28=N27+NUMNP
C
        CALL PRELIM(A(N1),A(N2),A(N3),A(N4),A(N5),A(N6))
        IF(N28 .LE. NFIELD) GO TO 100
C
C
C
        WRITE(6,1002) N28
C
C***********************************************************************
C       N28 IS TOTAL SPACE REQUIRED.  IF NFIELD, THE INITIALLY  ASSIGNED
C       SPACE, IS NOT ENOUGH, STOP THE PROGRAM.
C***********************************************************************
C
        STOP
  100 CONTINUE
C
        CALL PLAST(A(N1),A(N2),A(N3),A(N4),A(N5),A(N6),A(N7),A(N8),A(N9),
     1A(N10),A(N11),A(N12),A(N13),A(N14),A(N15),A(N16),A(N17),A(N18),
     2A(N19),A(N20),A(N21),A(N22),A(N23),A(N24),A(N25),A(N26),A(N27)
     3, NQ,NEL,FLIMIT,ITCONT,ACOEF,NDEX)
C
 1000 FORMAT(12A6)
 1001 FORMAT(///* THE DIMENSION OF THE ARRAY (A) IS TOO SMALL*/
     1* THE SIZE OF THE ARRAY (A) MUST BE *, I7)
 1002 FORMAT(//* THE NECCESSARY SIZE OF THE ARRAY (A) IS*, I7)
 1003 FORMAT(6I5,F10.0)
 1004 FORMAT(5F10.0)
 1005 FORMAT(4I5,F10.0)
C
        STOP
        END
```

```
        SUBROUTINE PRELIM(R,Z,UR,UZ,CODE,SLOP)
        COMMON/GENCON/NUMNP,NUMEL,HED(12),DLL,NEQ,NFORM,YIELD,TEST,ITER,
     1NREAD,NPUNCH,NPRINT,PVALUE,T,MBAND,PNRAD,RADIUS,FRITNP,FRITND,
```

```
      2ECONST,PNHED,RTHET,DIERAD,TCONTC,TDIST,RHECE
C
C************************************************************************MAINO
C     READ AND PRINT OF CONTROL INFORMATION AND MATERIAL PROPERTIES    MAIN0013
C************************************************************************MAIN0014
C
C
      DIMENSION R(1),Z(1),CODE(1),UR(1),UZ(1),SLOP(1)
C
   50 CONTINUE
      WRITE (6,2000) HED,NUMNP,NUMEL
      CALL HARD(0.,YIELD)
      WRITE(6,2010) YIELD
      WRITE(6,2011)
      WRITE(6,2012)RADIUS,PNRAD,DIERAD,RTHET,FRITNP,FRITND
      WRITE(6,2013)YVALUE,PRESTN,EXPNT,PRESTS
      WRITE(6,2014)ECONST
      WRITE(6,1009) ITER
C
C************************************************************************MAIN0030
C     READ AND PRINT OF NODAL POINT DATA                               MAIN0031
C************************************************************************MAIN0032
C
C
      L=0                                                              MAIN0034
      IF(NPPINT.EQ.0) GO TO 60
      WRITE (6,1114)
      WRITE (6,2004)                                                   MAIN0033
   60 READ  (5,1002) N,CODE(N),R(N),Z(N),UR(N),UZ(N),SLOP(N)
C
      NL=L+1                                                           MAIN0036
      ZX=N-L                                                           MAIN0037
      IF(L .EQ. 0) GO TO 70
      DR=(R(N)-R(L))/ZX                                                MAIN0038
      DZ=(Z(N)-Z(L))/ZX                                                MAIN0039
      DUR=(UR(N)-UR(L))/ZX
      DUZ=(UZ(N)-UZ(L))/ZX
      DS=(SLOP(N)-SLOP(L))/ZX
   70 L=L+1                                                            MAIN0041
C
      IF(N-L) 100,90,80                                                MAIN0042
C
   80 CODE(L)=0.0                                                      MAIN0043
      R(L)=R(L-1)+DR                                                   MAIN0044
      SLOP(L)=SLOP(L-1)+DS
      Z(L)=Z(L-1)+DZ
      UR(L)=UR(L-1)+DUR
      UZ(L)=UZ(L-1)+DUZ
      GO TO 70                                                         MAIN0049
C
   90 IF(NUMNP-N) 100,110,60
  100 WRITE (6,2009) N                                                 MAIN0052
      CALL EXIT                                                        MAIN0053
  110 CONTINUE                                                         MAIN0054
C
      WRITE (6,2002) (K,CODE(K),R(K),Z(K),UR(K),UZ(K),SLOP(K),K=1,NUMNP)
C
C
      NEQ=3*NUMNP
      WRITE(6,1122) NEQ,MBAND
C
C************************************************************************MAIN0119
 1002 FORMAT(I5,F5.0,5F10.0)
 1003 FORMAT(16I5)
 1004 FORMAT(I8,2I11,2F10.6)
 1005 FORMAT(2I5,4F10.0)
 1006 FORMAT(// * THE NODAL POINTS AT WHICH FORCE CALCULATIONS ARE DESIR
     1ED* // 20I5)
 1007 FORMAT(1H1,15X, 39H LINEARLY DISTRIBUTED BOUNDARY STRESSES/
     1 / 60H    NODE I   NODE J   PRESSURE I   PRESSURE J        SHEAR I
     2 14H        SHEAR J)
 1008 FORMAT(2I9,4E15.5)
 1009 FORMAT(///* MAXIMUM NUMBER OF ITERATIONS ALLOWED FOR EACH INCREMEN
     1T =*, I3)
 1114 FORMAT(1H1, 39H NODAL POINT INFORMATION BEFORE SCALING//)
 1122 FORMAT(/// * NUMBER OF EQUATIONS  =*, I4/
     1             * BANDWIDTH             =*, I4/
     2             * DIAGONAL ELEMENTS     =*, I4 )
 2000 FORMAT (1H  12A6/
     1 30H0 NUMBER OF NODAL POINTS------ I3 /                          MAIN0127
     2 30H0 NUMBER OF ELEMENTS---------- I3 /)
 2002 FORMAT (I12,F12.2,2F12.3,3E24.7)
 2003 FORMAT (I113,4I6,1I12)                                           MAIN0137
 2004 FORMAT (/ * NODAL POINT        TYPE  R-ORDINATE  Z-ORDINATE  R LO
     1AD OR DISPLACEMENT  Z LOAD OR DISPLACEMENT          BETA-SLOPE *)
```

```
 2005 FORMAT(//,*FORCES SPECIFIED AT NODAL POINT*,//,
     1 * NODAL PT.   ELEMENT1   ELEMENT2    PRESSURE        SHEAR*,/)
 2009 FORMAT (26HONODAL POINT CARD ERROR N= I5)                          MAIN0145
 2010 FORMAT(// * INITIAL YIELD STRESS = *, F15.7//)
 2011 FORMAT(///* SPECIFICATIONS OF THE PROBLEM*//)
 2012 FORMAT(* RADIUS OF PLANK IS       *,F10.5/
     1        * RADIUS OF PUNCH HEAD     *,F10.5/
     2        * RADIUS OF DIE PROFILE    *,F10.5/
     3        * DISTANCE FROM POLE TO THROAT*,F10.5/
     4        * FRICTION COEFFICIENT OVER PUNCH HEAD*,F10.5/
     5        * FRICTION COEFFICIENT OVER DIE*,F10.5/
     6        * WORKHARDENING CHARACTERISTICS*/)
 2013 FORMAT(/*STRESS=*,F10.5,*TIMES(*,F10.5,*+ STRAIN)EXP(*,F5.3,
     1*)+*,F10.3//)
 2014 FORMAT(///* STEP SIZE IS*,F10.5/)
      RETURN
      END


      SUBROUTINE PLAST(R,Z,UR,UZ,CODE,SLOP,YY,YX,SPHI,CPHI,DL,STS,TEPS,
     1EPS,B,A,THICK,ALPHA,GAMMA,ETA,FRNFCE,PHI,FF,TOUCH,CONTAC,UUR,UUZ,
     2NQ,NEL,FLIMIT,ITCONT,ACCEF,NDEX)
C
C***********************************************************************
C     PLAST IS THE CONTROLLING SUBROUTINE
C***********************************************************************
C
      COMMON/GENCON/NUMNP,NUMEL,HED(12),DLL,NEG,NFORM,YIELD,TEST,ITER,
     1NREAD,NPUNCH,NPRINT,EVALUE,T,MBAND,PNRAD,RADIUS,FRITNP,FRITND,
     2ECONST,PNHED,RTHRT,DIERAD,TCONTC,TDIST,BHFCE
      COMMON/CONQUAD/SS(4),WT(4),D(2,2),SOFT1
      COMMON/ATOUCH/NTOUCH,NDIE,NCONTC
C
      DIMENSION R(1),Z(1),UR(1),UZ(1),CODE(1),SLOP(1),YY(1),YX(1),
     1TEPS(4,1),B(1),A(NQ,1),THICK(1),CPHI(1),SPHI(1),DL(1),EPS(4,1),
     2ALPHA(1),GAMMA(1),ETA(1),FRNFCE(1),PHI(1),FF(1),CONTAC(1),TOUCH(1)
     3,STS(3,1),UUR(1),UUZ(1)
C
C***********************************************************************
C     THE FIRST NODE IS LOCATED AT THE RIM OF THE BLANK
C     AND THE POLE IS THE LAST NODE
C***********************************************************************
C
C
C***********************************************************************
C     A1,A2 ARE CONSTANTS RELATED TO DETERMINATION OF ACCEF
C     PNHED=PRESENT POSITION OF PUNCH HEAD
C     NTOUCH=FIRST NODAL POINT IN CONTACT WITH PUNCH HEAD
C     NCHECK=NUMBER OF NODES FOR WHICH THE CONTACTING WILL BE CHECKED
C     TCHCOF=0, IF BOUNDARY IS TO ADVANCE
C           =1, OTHERWISE
C***********************************************************************
C
      PNHED=Z(NUMNP)
      NCHECK=NUMEL/10
      NUM1=NUMNP-1
      NTOUCH=NUMNP
      TCHCOF=0.0
      SFACT=1.0
      TEST=0.0
      TEST=1.0
C
      DO 350 N=1,NUMNP
      IF(R(N) .GE. (RTHRT+DIERAD))NDIE=N
  350 IF(R(N) .GT. RTHRT)NTHRT=N
      NCONTC=NDIE
C
      SS(1)=0.8611363116
      SS(2)=0.3399810436
      SS(3)=-SS(1)
      SS(4)=-SS(2)
C
      DO 442 N=1, NUMEL
      FRNFCE(N)=0.0
      THICK(N)=T
      DO 442 I=1, 4
  442 TEPS(I,N)=0.
C
C***********************************************************************
C     TO HANDLE THE INFINITE WORKHARDENING RATE INITIALLY VERY
C     SMALL VALUE IS ASSIGNED TO THE EFFECTIVE STRAIN
C***********************************************************************
```

```
C
      DO 450 N=1,NUMEL
  450 TEPS(4,N)=0.0001
C
      FRNFCE(NTOUCH)=.001
      FRNFCE(1)=BHFCE
      DO 360 N=NDIE,NCONTC
  360 FRNFCE(N)=.0001
C
      DPNSTR=0.1
      DPNHED=UZ(NUMNP)
      ESTAR=1.0
C
C***************************************************************************
C     IF THE COMPUTATION IS INTERRUPTED AFTER A NUMBER OF STEPS
C     AND RESTARTED, THEN NECESSARY DATA NEED BE FEED
C***************************************************************************
C
      IF(NREAD .LE. 0) GO TO 440
      READ(5,1017)(UR(I),UZ(I),CODE(I),I=1,NUMNP)
      READ(5,1017) ( R(I),  Z(I), I=1, NUMNP)
      READ(5,1017)((TEPS(I,N),I=1,4), N=1, NUMEL)
      READ(5,1017)(THICK(N),N=1,NUMEL)
      READ(5,2223)PNHED,NTOUCH,TCHCOF,EFACT,NCONTC
      READ(5,235)(FRNFCE(N),N=1,NUM1)
      READ(5,233)ESTAR,DPNSTR,DPNHED,TEST
  440 CONTINUE
C
      NSTEP=0
      TCONT2=TCONTC
      TOUCH2=TDIST
C
 2100 NSTEP=NSTEP+1
C
C***************************************************************************
C     BOUNDARY NODE NDIE IS UPDATED AFTER EVERY STEP
C***************************************************************************
C
      DO 351 N=1,NUMNP
  351 IF(R(N) .GE. (RTHRT+DIERAD))NDIE=N
      IF(FRNFCE(NDIE) .EQ. 0.)FRNFCE(NDIE)=.0001
C
      TCONT3=TCONT2
      TOUCH3=TOUCH2
C
      DO 247 I=1,NCHECK
      CONTAC(I)=0.0
  247 TOUCH(I)=0.
      IF(TEST .EQ. 0.)GO TO 2101
C
C***************************************************************************
C     DPNHED=ASSIGNED INCREMENT OF PUNCH HEAD TRAVEL
C     ESTAR=ADJUSTING FACTOR
C***************************************************************************
C
      PNHED=PNHED+DPNHED
C
      DO 445 I=1,NUMNP
      UR(I)=UR(I)*ESTAR
      UZ(I)=UZ(I)*ESTAR
  445 CONTINUE
C
 2101 CONTINUE
      IF(TCHCOF .EQ. 1.)GO TO 219
      NTOUCH=NTOUCH-1
      NTCP1=NTOUCH+1
      FRNFCE(NTOUCH)=FRNFCE(NTCP1)/3.
  219 CONTINUE
  446 CONTINUE
C
C***************************************************************************
C     UPDATING OF THE BOUNDARY CONDITION
C***************************************************************************
C
      DO 100 I=1,NUMNP
      CODE(I)=0.0
      IF(I .GE. NTOUCH)CODE(I)=4.0
      IF(I .GT. NDIE .AND. I .LE. NCONTC) CODE(I)=4.0
      IF(I .LE. NDIE) CODE(I)=2.0
  100 CONTINUE
      CODE(NUMNP)=3.0
      CODE(1)=2.0
      IF(NDEX .EQ. 2) CODE(1)=3.0
C
      NTCHM1=NTOUCH-1
```

```
C
C
C*****************************************************************************
C     COMPUTE THE YIELD STRESS AND THE WORKHARDENING RATE
C*****************************************************************************
C
      DO 220 N=1, NUMEL
      CALL HARD(TEPS(4,N),YY(N))
      CALL HARD2(TEPS(4,N),YX(N))
  220 CONTINUE
C
C
C
C
C
      WRITE(6,1007) NSTEP
C
      CONV=111111.
C
  659 CONTINUE
      WRITE(6,1031)
C
C*****************************************************************************
C     DETAIL OF THE PRESENT CONFIGURATION
C     SPHI=SINE OF ANGLE PHI
C     CPHI=COSINE OF ANGLE PHI
C     DL=ELEMENT LENGTH
C*****************************************************************************
C
      DO 690 N=1, NUMEL
      NP1=N+1
      DR=R(N)-R(NP1)
      DZ=Z(NP1)-Z(N)
      DL(N)=SQRT(DR*DR+DZ*DZ)
      SPHI(N)=DR/DL(N)
      CPHI(N)=DZ/DL(N)
      PHI(N)=ASIN(SPHI(N))*180./3.14156
      WRITE(6,1030)N,PHI(N),THICK(N),DL(N)
  690 CONTINUE
C
      RP1=RVALUE+1.
      RCONST=3.*RP1/( 2.*(1.+RVALUE+RVALUE) )
      D(1,1)=RP1*RCONST
      D(1,2)=RVALUE*RCONST
      D(2,1)=D(1,2)
      D(2,2)=D(1,1)
      CLAMDA=RP1*RP1/(1.+RVALUE+RVALUE)
      POISON=RVALUE/RP1
C
      K=0
 2001 CONTINUE
      K=K+1
C
C
      CALL STIFF(R,Z,UR,UZ,CODE,SLOP,YY,YX,SPHI,CPHI,DL,EPS,
     1THICK,ALPHA,GAMMA,ETA,ERNECE,FF,A,B,NQ)
C
C*****************************************************************************
C     INTRODUCTION OF BOUNDARY CONDITION
C*****************************************************************************
C
C
      CALL MODIFY(CODE,A,B,ALPHA,GAMMA,ETA,NUMNP,NEQ,MBAND,ERNECE)
C
C     BANDED SYMMETRIC SOLUTION
C
      CALL TRIA(NEQ,MBAND,A)
      CALL BACKS(NEQ,MBAND,A,B)
C
C
C*****************************************************************************
C     PERTURBATION OF UR IS COMPUTED FROM PERTURBATION OF UZ FOR NODES
C*****************************************************************************
C
C
C
C
      DO 101 N=1,NUMNP
      IR=3*N-2
      IZ=IR+1
      IF(CODE(N) .EQ. 4.0) B(IR)=B(IZ)/ALPHA(N)+GAMMA(N)
  101 CONTINUE
C*****************************************************************************
C     TO OBTAIN AN EFFICIENT CONVERGENCE ACCEF IS COMPUTED
C     THE ACCELERATING COEFFICIENT IS DETERMINED IN SUCH A WAY
```

```
C      PERTURBATIONAL TERM TIMES ACCELERATING COEFFICIENT IS NEVER
C      GREATER THAN THE INITIAL VALUE, BUT BE A FRACTION OF IT.
C      E.G. A2=2 MEANS HALF
C*************************************************************************
       A1=10.
       A2=2.
       CONCOF=0.0
       DO 103 I=1,NUMNP
       IF(ABS(UR(I)) .LT. .0000001)UR(I)=0.
       IF(ABS(UZ(I)) .LT. .0000001)UZ(I)=0.
       UUR(I)=UR(I)
       UUZ(I)=UZ(I)
       IZ=3*I-1
       IR=IZ-1
       IF(B(IR) .EQ. 0.)GO TO 102
       COF1=ABS(UR(I)/B(IR))
   102 IF(B(IZ) .EQ. 0.)GO TO 103
       COF2=ABS(UZ(I)/B(IZ))
       A1=AMIN1(A1,COF1,COF2)
   103 CONTINUE
C
C
   105 CONTINUE
       IF(CONCOF .EQ. 1.)A2=A2*5.
       IF(A1 .EQ. 1.0 .AND. A2 .GT. 10.)COF1=5.
       IF(COF1  .EQ. 0.)COF1=2.
       A1=COF1/A2
       IF(A1 .LT. .00001)GO TO 2300
       IF(A1 .GT. 1.0) A1=1.0
       ACOEF=A1
       WRITE(6,104)A1
C
C
C*************************************************************************
C      OBTAIN NEW VALUE
C*************************************************************************
C
       DO 130 I=1, NUMNP
       IZ=3*I-1
       IR=IZ-1
       IS=IZ+1
       UR(I)=UUR(I)
       UZ(I)=UUZ(I)
       UR(I)=UR(I)+B(IR)*ACOEF
       UZ(I)=UZ(I)+B(IZ)*ACOEF
       SLOP(I)=SLOP(I)+B(IS)*ACOEF
   130 CONTINUE
C
       WRITE(6,1016) K
       WRITE(6,1006) K
C
C
C*************************************************************************
C      COMPUTE NORM OF ERROR AND NORM OF SOLUTION.
C*************************************************************************
C
       ENORM = 0.
       SNORM = 0.
       DO 134 I=1, NUMNP
       IZ=3*I-1
       IR=IZ-1
       IS=IZ+1
       ENORM = ENORM + B(IR)*B(IR) + B(IZ)*B(IZ) + B(IS)*B(IS)
       SNORM = SNORM + UR(I)*UR(I) + UZ(I)*UZ(I) + SLOP(I)*SLOP(I)
   134 CONTINUE
C
C
       ENORM = SQRT(ENORM)
       SNORM = SQRT(SNORM)
       ESNORM=ENORM/SNORM
       WRITE(6,1015) SNORM,ENORM,ESNORM
       DO 776 I=1, NUMNP
       IZ=3*I-1
       IR=IZ-1
       IS=IZ+1
       WRITE(6,1002) I,B(IR),B(IZ),B(IS),UR(I),UZ(I),SLOP(I),R(I),Z(I)
   776 CONTINUE
C
   131 CONTINUE
C
C
C
C*************************************************************************
C      COMPUTE STRAIN FROM THE NEW GUESS.
C      EPS(1,N)=INCREMENT OF MERIDIAN STRAIN
```

```
C      EPS(2,N)=INCREMENT OF TANGENTIAL STRAIN
C      EPS(3,N)=INCREMENT OF THICKNESS STRAIN
C*********************************************************************
C
C
C
       DO 800 N=1,NUMEL
       NP1=N+1
       DLL=DL(N)
       SPH=SPHI(N)
       CPH=CPHI(N)
       AU=UR(N)+UR(NP1)
       AR=R(N)+R(NP1)
       DR=R(N)-R(NP1)
       DZ=Z(NP1)-Z(N)
       DU=UR(N)-UR(NP1)
       DW=UZ(NP1)-UZ(N)
       EX1=1.+2.*DR*DU/DLL/DLL+2.*DZ*DW/DLL/DLL+(DU*DU+DW*DW)/DLL/DLL
C
       EPS(1,N)=ALOG(EX1)/2.
       EPS(2,N)=ALOG(1.+AU/AR)
       EPS(3,N)=-EPS(1,N)-EPS(2,N)
  800  CONTINUE
       TEST=0.0
C
C
C
C*********************************************************************
C      COMPUTE INCREMENT OF EFFECTIVE STRAIN
C*********************************************************************
C
       WRITE(6,1026) NSTEP
       DO 222 N=1, NUMEL
       NP1=N+1
       IF(CODE(N) .EQ. 3.0 .AND. CODE(NP1) .EQ. 3.0)GO TO 222
       ES=EPS(1,N)
       ET=EPS(2,N)
       RBAR=RP1*(ES*ES+ET*ET) + 2.*RVALUE*ES*ET
C
       EPS(4,N)=SQRT(2.*RCONST*RBAR/3.)
C
C
       STS(1,N)=CLAMDA*(ES + POISON*ET)*YY(N)/EPS(4,N)
       STS(2,N)=CLAMDA*(ET + POISON*ES)*YY(N)/EPS(4,N)
C
C*********************************************************************
C      COMPUTE STRESS DISTRIBUTION
C      STS(1,N)=MERIDIAN STRESS
C      STS(2,N)=CIRCUMFERENTIAL STRESS
C      STS(3,N)=EFFECTIVE STRESS
C*********************************************************************
C
       IF(ESNORM .LT. FLIMIT)TEST=1.0
       ES=STS(1,N)
       ET=STS(2,N)
       EFSTRS=ES*ES+ET*ET-2.*POISON*ES*ET
       STS(3,N)= SQRT(EFSTRS)
C
       WRITE(6,1003)N,(EPS(I,N),I=1,4)
  222  CONTINUE
       WRITE(6,1027)
       DO 430 N=1,NUMEL
  430  WRITE(6,2251)N,(STS(I,N),I=1,3)
C
C*********************************************************************
C      CHECK WHETHER ACOEF IS TOO LARGE TO CAUSE A PHYSICALLY
C      UNACCEPTABLE SOLUTION. WHENEVER COMPUTED MERIDIAN STRESS BECOMES
C      NEGATIVE ADJUST ACOEF VALUE)
C*********************************************************************
C
       CONCOF=0.0
       STS1=0.0
       DO 431 N=1,NUMEL
       NP1=N+1
       IF(CODE(N) .EQ. 3.0 .AND. CODE(NP1) .EQ. 3.0)STS(1,N)=0.
  431  IF(STS(1,N) .LT. STS1)CONCOF=1.0
C
C*********************************************************************
C      CHECK WHETHER (ERROR NORM)/(SOLUTION NORM) IS LESS THAN FLIMIT
C      IF YES, THE SOLUTION IS FINAL
C*********************************************************************
C
C
C
       IF(ESNORM .LT. FLIMIT)GO TO 438
```

```
C
      IF(K .GE. ITER)GO TO 2207
C
C
      DO 1900 N=1,NUMEL
      IF(EPS(4,N) .GT. .000001)GO TO 1900
      NP1=N+1
      DUM=UR(N)*UR(NP1)
      CODE(NP1)=3.0
      CODE(N)=3.0
 1900 CONTINUE
C
      GO TO 2001
 2000 CONTINUE
 2200 CONTINUE
C
  438 CONTINUE
      IF(ESNORM .GT. FLIMIT)GO TO 777
C
C
      WRITE(6,2800)
C
C     EXAMINATION ON BOUNDARY ASSUMPTIONS
C*****************************************************************************
C
      DO 250 I=1,NCHECK
      N=NTOUCH-I
      IF(N .LE. 1)GO TO 250
      TOUCH(I)=(Z(N)+UZ(N)+PNRAD-PNHED)*(Z(N)+UZ(N)+PNRAD-PNHED)+(R(N)
     1+UR(N))*(R(N)+UR(N))
      TOUCH(I)=SQRT(TOUCH(I))-PNRAD
      WRITE(6,2900)N,TOUCH(I)
  250 CONTINUE
      WRITE(6,265)
C
C
C*****************************************************************************
C     COMPUTE THE DISTANCES OF THE FREE NODE FROM THE DIE PROFILE
C*****************************************************************************
C
      RDIE=RTHRT+DIERAD
      DO 260 I=1,NCHECK
      N=NCONTC+I
      CONTAC(I)=(DIERAD-Z(N)-UZ(N))*(DIERAD-Z(N)-UZ(N))+(RDIE-R(N)-UR(N)
     1)*(RDIE-R(N)-UR(N))
      CONTAC(I)=SQRT(CONTAC(I))-DIERAD
  260 WRITE(6,270)N,CONTAC(I)
C
      IF(ABS(TOUCH(1)) .LT. .0001)TOUCH(1)=0.
      IF(ABS(CONTAC(1)) .LT. .0001)CONTAC(1)=0.
C
C*****************************************************************************
C     CHECK ON BOUNDARY OVER PUNCH HEAD
C*****************************************************************************
C
      IF(TOUCH(1) .GE. 0.)GO TO 3000
C
      WRITE(6,3100)
C
C*****************************************************************************
C     NODE AT NCHM1 IS INSIDE PUNCH, COMPUTE AGAIN
C*****************************************************************************
C
      TCHCOF=0.0
      TINSDE=0.0
      GO TO 2101
C
C
 3000 CONTINUE
      TCHCOF=1.0
C
C*****************************************************************************
C     CHECK ON BOUNDARY OVER DIE PROFILE
C*****************************************************************************
C
      IF(CONTAC(1) .GE. 0.)GO TO 3500
      WRITE(6,3400)
C
C*****************************************************************************
C     NODE AT NCONTC+1 IS INSIDE DIE, COMPUTE AGAIN
C*****************************************************************************
C
      NCONTC=NCONTC+1
      NC1=NCONTC-1
      FRNFCE(NCONTC)=FRNFCE(NC1)
```

```
C
      GO TO 2101
 3500 CONTINUE
      IF(NDEX .EQ. 2)GO TO 3501
      IF((R(NDIE)+UR(NDIE)) .GT. RDIE)GO TO 3501
      NDIE=NDIE-1
      WRITE(6,3450)
C
C*****************************************************************************
C     NDIE NODE HAS BEEN BROUGHT INTO CONTACT WITH DIE PROFILE
C*****************************************************************************
C
      FRNFCE(NDIE)=0.0001
      GO TO 2101
C
 3501 CONTINUE
      TINSDE=1.0
      TEST=1.0
C
C*****************************************************************************
C     COMPUTATION OF FRICTION COEFFICIENT
C*****************************************************************************
C
      MUDEX=0
      PPNHED=PNHED
      WRITE(6,233)(FRNFCE(N),N=1,NUM1)
      WRITE(6,231)
C
      DO 230 I=1,NUM1
      IZ=3*I-1
      IR=IZ-1
      IF(CODE(I) .EQ. 3.0)GO TO 230
      IF(I .GE. NTOUCH)GO TO 280
      IF(I .LE. NCONTC .AND. I .GT. NDIE)GO TO 281
      IF(I .LT. NTOUCH .AND. I .GT. NCONTC)GO TO 230
      IF(NDEX .EQ. 2)GO TO 230
      IF(I .LE. NDIE)GO TO 282
C
  280 CONTINUE
      DUM1=(Z(I)+UZ(I)+PNRAD-PPNHED)/PNRAD
      DUM2=(R(I)+UR(I))/PNRAD
      FRITN=FRITNP
      IF(FRITN .EQ. 0.)GO TO 230
      GO TO 283
C
  281 DUM1=DIERAD-Z(I)-UZ(I)
      DUM2=RDIE-R(I)-UR(I)
      FRITN=FRITND
      IF(FRITN .EQ. 0.)GO TO 230
      GO TO 283
C
  282 DUM1=1.
      DUM2=0.
      FRITN=FRITND
      IF(FRITN .EQ. 0.)GO TO 230
C
  283 CONTINUE
      PN=FF(IZ)*DUM1+FF(IR)*DUM2
      PT=FF(IZ)*DUM2-FF(IR)*DUM1
      IF(PN .EQ. 0.)GO TO 230
      XMU=PT/PN
      WRITE(6,232)I,XMU
      XMU=XMU/FRITN
      IF(XMU .GT. 1.02 .OR. XMU .LT. .98)MUDEX=1
      FRNFCE(I)=FRNFCE(I)/XMU
      IF(I .EQ. NCONTC)PPN=-PN
  230 CONTINUE
  234 CONTINUE
C
C
C
C*****************************************************************************
C     MUDEX=0, IF FRICTION CONDITION IS SATISFIED
C          =1, OTHERWISE
C*****************************************************************************
C
C
      IF(MUDEX .EQ. 1)TCHCOF=1.0
      IF(MUDEX .EQ. 1)GO TO 2001
C
      ICHECK=0
C
C*****************************************************************************
C     GENERALIZED NODAL FORCE NORMAL TO THE PUNCH IS COMPUTED
C     TO CHECK WHETHER THE BOUNDARY IS ASSUMED TO MOVE TOO FAST.
```

```
C***********************************************************************
C
      IF(CODE(NTOUCH) .EQ. 3.0)GO TO 500
C
      NZZ=3*NTOUCH-1
      IF(FF(NZZ) .GT. 0.) GO TO 500
      IF(ABS(FF(NZZ)) .LT. .000001 .OR. TINSDE .EQ. 0.)GO TO 500
      TDIST=TOUCH3*.8
      WRITE(6,510)TDIST
      NTOUCH=NTOUCH+1
      TCHCOF=1.0
      ICHECK=1
  500 CONTINUE
C
C
C
C***********************************************************************
C     GENERALIZED NODAL FORCE NORMAL TO DIE IS COMPUTED.
C***********************************************************************
C
      IF(CODE(NCONTC) .EQ. 3.0)GO TO 550
      IF(NCONTC .EQ. NDIE)GO TO 550
      IF(PPN .GT. -.000001)GO TO 550
C
      TCONTC=TCONT3*.8
      WRITE(6,549)TCONTC
      NCONTC=NCONTC-1
      ICHECK=1
  550 CONTINUE
C
C***********************************************************************
C     ICHECK=1. IF BOUNDARY ASSUMPTION NEEDS TO BE MODIFIED.
C***********************************************************************
C
      IF(ICHECK .EQ. 1)GO TO 446
      NNCONC=NCONTC
C
C***********************************************************************
C     MAKE BOUNDARY ASSUMPTION FOR NEW STEP BASED UPON
C     THE DISTANCE AWAY FROM PUNCH OF DIE
C***********************************************************************
C
      DO 560 I=1,NCHECK
      IF(CONTAC(I)-TCONTC)561,562,562
  561 NCONTC=NNCONC+I
      TCONT2=CONTAC(I)
  560 FRNFCE(NCONTC)=FRNFCE(NNCONC)
  562 CONTINUE
C
C
C
C
C
      NNTCH=NTOUCH
      DO 240 I=1,NCHECK
      IF(TOUCH(I)-TDIST)245,246,246
  245 NTOUCH=NNTCH-I
      TOUCH2=TOUCH(I)
      FRNFCE(NTOUCH)=FRNFCE(NNTCH)
  240 CONTINUE
  246 TCHCOF=1.0
C
C***********************************************************************
C     COMPUTE TOTAL STRAIN
C***********************************************************************
C
      WRITE(6,1043)
      DO 444 N=1,NUMEL
      DO 443 I=1, 4
  443 TEPS(I,N)=TEPS(I,N)+EPS(I,N)*TEST
      IF(ESNORM .LT. FLIMIT)THICK(N)=THICK(N)*EXP(EPS(3,N)*TEST)
      WRITE(6,1003)N,(TEPS(I,N),I=1,4)
  444 CONTINUE
C
C
C
C
C***********************************************************************
C     COMPUTE DPNHED FOR NEXT STEP WHICH WOULD GIVE THE INCREMENT
C     OF MAXIMUM EFFECTIVE STRAIN APPROXIMATELY EQUAL TO PRESET VALUE.
C***********************************************************************
C
      EMAX=0.0
      DO 775 N=1,NUMEL
  775 IF(EPS(4,N) .GT. EMAX) EMAX=EPS(4,N)
```

```
      EFACTA=ECONST/EMAX
      IF(NSTEP .LE. 2  .AND.  ITCONT  .EQ.  0)GO TO 778
      DPNHED=2./EFACTA/UZ(NUMNP)-1./EFACT/DPNSTR
      DPNHED=1./DPNHED
      ESTAR=DPNHED/UZ(NUMNP)
      DPNSTR=UZ(NUMNP)
  778 CONTINUE
      EFACT=EFACTA
C
C
C
  777 CONTINUE
      IF(ESNORM .GT. FLIMIT)TCHCOF=1.0
C
C*****************************************************************************
C     NEW CONFIGURATION
C*****************************************************************************
C
      DO 439 I=1, NUMNP
      IZ=3*I-1
      IR=IZ-1
      R(I)=R(I)+UR(I)*TEST
      Z(I)=Z(I)+UZ(I)*TEST
  439 CONTINUE
C
C
C
C*****************************************************************************
C     PUNCH THE SOLUTION
C*****************************************************************************
C
      IF(NPUNCH .EQ. 0) GO TO 310
 2307 CONTINUE
      PUNCH 1017,(UR(I),UZ(I),CODE(I),I=1,NUMNP)
      PUNCH 1017, (R(I),Z(I),I=1, NUMNP)
      PUNCH 1017, ( (TEPS(I,N),I=1,4), N=1, NUMEL)
      PUNCH 1017,(THICK(N),N=1,NUMEL)
      PUNCH 2223,PNHED,NTOUCH,TCHCOF,EFACT,NCONTC
      PUNCH 235,(FRNFCE(N),N=1,NUM1)
      PUNCH 233,ESTAR,DPNSTR,DPNHED,TEST
  310 CONTINUE
C
      IF(ESNORM .GT. FLIMIT)GO TO 2300
C
C
      WRITE(5,1040)
      DO 849 I=1,NUMNP
      IR=3*I-2
      IZ=IR+1
      IL=IZ+1
      WRITE(5,1041)(I,FF(IR),FF(IZ),FF(IL))
  849 CONTINUE
C
C
C*****************************************************************************
C     COMPUTE THE PUNCH LOAD FROM ENERGY BALANCE
C*****************************************************************************
C
      SUMF=0.
      DO 850 I=2,NUMNP
      IZ=I*3-1
      IR=IZ-1
  850 SUMF=SUMF+FF(IZ)*UZ(I)+FF(IR)*UR(I)
      SUMF=SUMF/UZ(NUMNP)/TEST*2.*3.14
C
      WRITE(6,1042)SUMF
      IF(NSTOP .EQ. 1) GO TO 2300
      WRITE(6,1028)FPNHED,NNTCH
      WRITE(6,1029)EMAX,UZ(NUMNP),EFACT
      IF(NSTEP .LT. NFORM) GO TO 2100
 2300 CONTINUE
 2301 CONTINUE
C
C
 1002 FORMAT(I5,3F13.7,5X,3F13.7,5X,3F13.7)
 1003 FORMAT(I7,11F11.6)
 1004 FORMAT(16I5)
 1005 FORMAT(1H1,* STRAIN-STRESS SOLUTION AT  STEP  NUMBER  =*,I4//
     1 * EL. NO...R-STRAIN...Z-STRAIN..TH-STRAIN...RZ-STRAIN...EF-STRAIN
     2...R-STRES...Z-STRES...TH-STRES...RZ-STRES..EF-STRES..AVG-STRES..*
     3)
 1006 FORMAT(///  30X, * DISPLACEMENT SOLUTION AT ITERATION NUMBER =*,I4
     1///  20X, * PERTURBED*, 26X, * TOTAL*, 20X, * DEFORMED COORD*/
     2/   *   NP          DU            DW              DBETA          U
     3       W                  BETA             R              Z*)
```

```
 1007 FORMAT(1H1,70X,*ITERATION PROCESS FOR STEP*,I4)
 1008 FORMAT(      60X, * TOTAL R-LOAD =*, F12.7
     1           /  60X, * TOTAL Z-LOAD =*, F12.7
     2           /  60X, * TOTAL M-LOAD =*, F12.7)
 1010 FORMAT(   // * NODAL POINT FORCE AT STEP =*, I4//
     1*.......N.P.........R-FORCE.......Z-FORCE......Z-STRESS ON DIE SUR
     2FACE...*)
 1011 FORMAT(I5,3F10.0)
 1012 FORMAT(I5,6F17.5)
 1015 FORMAT(60X,* VELOCITY CONVERGENCE*,/
     1                 60X, * NORM OF SOLUTION VECTOR =*, F13.8
     1           /  60X, * NORM OF   ERROR  VECTOR =*, F13.8
     2           /  60X, * FRACTIONAL  NORM          =*, F13.8)
 1016 FORMAT(             * DISPLACEMENT SOLUTION AT ITERATION NUMBER =*,I4)
 1017 FORMAT(8F10.7)
 1018 FORMAT(////* DOES NOT CONVERGE*//
     1* TRY AGAIN WITH DECELLERATION COEFFICIENT =ACOEF= LESS THAN*,
     2F8.3)
 1020 FORMAT(20F4.1)
 1025 FORMAT(4X,I5,3X,F12.6,10X,I5,3X,F12.6,10X,I5,3X,F12.6)
 2251 FORMAT(I5,4F20.7)
 1026 FORMAT(///*INCREMENTAL STRAIN-TOTAL STRAIN AT STEP NUMBER=*,I4//
     1*EL  NO....S-STRAIN......THE-STRAIN.......THI-STRAIN......EF-STRAN
     1*)
 1027 FORMAT(///*EL.  NO....S-STRESS....THE-STRESS....EF-STRESS....*)
 1042 FORMAT(*   PUNCH  FORCE=*,F15.7)
 1043 FORMAT(///*EL  NO....S-STRAIN......THE-STRAIN......THI-STRAIN....
     1EF-STRAIN...*)
 1041 FORMAT(5X,I10,5X,3F20.7)
 1040 FORMAT(//*      NO.  OF  NODE                       FORCE*)
  510 FORMAT(//*    NTOUCH IS FORCED TO TOUCH, COMPUTE  AGAIN
     1*,/*   TDIST=*,F10.7)
 1030 FORMAT(/I7,3F10.5)
 1031 FORMAT(*  GEOMETRY  OF  PROFILE*//
     1 *EL  NNO.....ANGLE.....THICKNESS.....*)
  104 FORMAT(//*      ACOEF  CALCULATED*,F10.7)
  265 FORMAT(//*  CHECKING  DISTANCE  AWAY  FROM  DIE*)
  270 FORMAT(I5,F20.7)
 2900 FORMAT(/I10,F20.7)
 2800 FORMAT(//*  CHECKING  DISTANCE  AWAY  FROM  PUNCH*,/
     1*  ELEM  NO.            TOUCH*/)
 3100 FORMAT(*  NODE  AT  NCHM1 IS INSIDE PUNCH, COMP  AGAIN*)
 3400 FORMAT( /* NODE  AT  NCONTC+1  IS  INSIDE  DIE, COMP  AGAIN*)
 3450 FORMAT(/* NDIE NODE HAS BEEN BROUGHT INTO CONTACT WITH CORNER*/)
  231 FORMAT(//*   NODAL  POINT          COEFFICIENT*,/)
  549 FORMAT(/*  NCONTC IS FORCED TO TOUCH, TCONTC=*,F10.7)
  232 FORMAT(I10,F10.5)
  233 FORMAT(5F15.7)
  235 FORMAT(4F15.7)
 2223 FORMAT(F15.7,I5,2F15.7,I5)
 1028 FORMAT(*   PUNCH  HEAD  DISPLACEMENT*,F10.5/*     NTOUCH=  *,I5)
 1029 FORMAT(///*   MAX  EFFECTIVE  STRAIN  INCREMENT**,F10.7/*  PUNCH
     1 HEAD INCREMENT=*,F10.7,/*  PUNCH  HEAD  ADJUSTING  FACTOR=*,F10.7
     2)
C
      RETURN
      END
```

```
      SUBROUTINE STIFF(R,Z,UR,UZ,CODE,SLOP,YY,YX,SPHI,CPHI,DL,EFS,
     1THICK,ALPHA,GAMMA,ETA,FRNFCE,FF,A,B,NO)
C
C*******************************************************************************
C      CALCULATION OF STIFFNESS MATRIX FOR ENTIRE SYSTEM
C*******************************************************************************
C
      COMMON/GENCON/NUMNP,NUMEL,HED(12),DLL,NEQ,NFORM,YIELD,TEST,ITER,
     1NREAD,NPUNCH,NPRINT,RVALUE,T,MBAND,PNEAD,RADIUS,FRITNP,FRITND,
     2ECONST,FNHED,FTHRT,DIERAD,TCONTC,TDIST,RHFCE
      DIMENSION R(1),Z(1),CODE(1),UR(1),UZ(1),SLOP(1),B(1),A(NO,1),
     1 EFS(4,1),RR(2),ZZ(2),UU(6),YY(1),THICK(1),DL(1),SPHI(1),CPHI(1),
     2 YX(1),ALPHA(1),GAMMA(1),ETA(1),FRNFCE(1),FF(1)
      COMMON/STFMAT/H(6),P(6,6),TEX,TEY,TEZ,THKL
      COMMON/CONQUAD/SS(4),WT(4),D(2,2),SOFT1
      COMMON/ATOUCH/NTOUCH,NDIE,NCONTC
C
C
C*******************************************************************************
C      INITIALIZE A AND B MATRIX FOR EQUATION AX=B
C      BECAUSE BANDED SYMMETRIC PROPERTY OF THE STIFFNESS MATRIX A.
C      THE STORAGE OF A IS IN A SQUARE MATRIX
C*******************************************************************************
```

```
C
      DO 50 N=1, NEQ
      B(N)=0.
      DO 50 M=1,MBAND
   50 A(N,M)=0.
      WT(1)=0.3478548451
      WT(2)=0.6521451549
      WT(3)=WT(1)
      WT(4)=WT(2)
C
C
C*****************************************************************************
C     CONSTRUCT P AND H AT ELEMENT LEVEL
C*****************************************************************************
C
      DO 1000 N=1, NUMEL
      NP1=N+1
      IF(CODE(N) .EQ. 3.0 .AND. CODE(NP1) .EQ. 3.0) GO TO 1000
      DLL=DL(N)
      SPH=SPHI(N)
      CPH=CPHI(N)
      RR(1)=R(N)
      ZZ(1)=Z(N)
      RR(2)=R(NP1)
      UU(1)=UR(N)
      UU(2)=UZ(N)
      UU(3)=SLOP(N)
      UU(4)=UR(NP1)
      UU(5)=UZ(NP1)
      UU(6)=SLOP(NP1)
      THKL=THICK(N)*DLL
      ZZ(2)=Z(NP1)
      YG=YX(N)
      YH=YY(N)
C
      CALL QUAD(RR,ZZ,UU,DLL,SPH,CPH,YG,YH)
C
C
C
C*****************************************************************************
C     PERFORM THE ASSEMBLY OPERATION. BECAUSE MATRIX A IS SYMMETRIC
C     ONLY UPPER HALF OF THE MATRIX IS CREATED.  AND THE STORAGE FOR
C     MATRIX A IS A SQUARE ARRAY BECAUSE OF BANDED SYMMETRIC PROPERTY
C*****************************************************************************
C
      DO 200 I=1, 6
      II=N*3 - 3 + I
      B(II)=B(II)+H(I)
      DO 200 J=1, 6
C
      JJ=N*3-3+J-II+1
      IF(JJ .LT. 1) GO TO 200
      A(II,JJ)=A(II,JJ)+P(I,J)
  200 CONTINUE
 1000 CONTINUE
C
C*****************************************************************************
C     TO HANDLE MIXED BOUNDARY CONDITION, FOLLOWING MATRICES ARE EVALUATED
C*****************************************************************************
C
      C=PNRAD-PNHED
      NMP1=NUMNP-1
      PPNRAD=PNRAD
      RDIE=RTHRT+DIERAD
C
      DO 1200 N=1,NMP1
      IF(CODE(N) .NE. 4.0)GO TO 1200
      DUM2=RDIE-R(N)-UR(N)
      DUM1=DIERAD-Z(N)-UZ(N)
      PNRAD=DIERAD
      IF(N .LE. NCONTC)GO TO 1100
      PNRAD=PPNRAD
      DUM2=R(N)+UR(N)
      DUM1=C+Z(N)+UZ(N)
C
 1100 CONTINUE
      ALPHA(N)=-DUM2/DUM1
      GAMMA(N)=(PNRAD*PNRAD-DUM2*DUM2-DUM1*DUM1)/2./DUM2
      ETA(N)=FRNFCE(N)*PNRAD/DUM2
      IF(N .GE. NTOUCH)GO TO 1200
      GAMMA(N)=-GAMMA(N)
      ETA(N)=-ETA(N)
 1200 CONTINUE
C
C*****************************************************************************
```

```
C      STORE GENERALIZED NODAL FORCE
C**********************************************************************
C
       NUM3=NUMNP*3
       DO 1300 I=1,NUM3
 1300  FF(I)=-B(I)
C
C
C
C
 1001  FORMAT(///,* THE DIAGONAL VECTOR OF MATRIX OF STIFFNESS*/)
 1002  FORMAT(12E11.3)
 1005  FORMAT(// 29H ELEMENT WITH NEGATIVE AREA =, I5)
C
       RETURN
       END



       SUBROUTINE QUAD(RR,ZZ,UU,DLL,SPH,CPH,S2,S1)
       COMMON/STFMAT/H(6),P(6,6),TEX,TEY,TEZ,THKL
       COMMON/CONQUAD/SS(4),WT(4),D(2,2),SQFT1
       DIMENSION RR(2),ZZ(2),UU(6),B(2,6),XX(6,6),BZERO(6),DB(2,6)
       COMMON/ISOTRY/RVALUE
C
       RC=(RR(1)+RR(2))/2.
C
       DO 2 I=1,6
       H(I)=0.
       DO 2 J=1,6
     2 P(I,J)=0.
       TEX=0.
       TEY=0.
       TEZ=0.
C
C
       DZ=ZZ(2)-ZZ(1)
       DR=RR(1)-RR(2)
       DU=UU(1)-UU(4)
       DW=UU(5)-UU(2)
       AU=UU(1)+UU(4)
       AR=RR(1)+RR(2)
C
       C1=2.*DR/DLL/DLL
       C2=2.*DU/DLL/DLL
       C3=2.*DZ/DLL/DLL
       C4=2.*DW/DLL/DLL
       C5=AU/AR/2.
       C6=1.+DR*C2+DZ*C4+(DU*DU+DW*DW)/DLL/DLL
       C7=2./DLL/DLL
       C8=2./AR/AR
       C9=1./SQRT(C6)/2.
       C10=C9/C6
       C11=C1+C2
       C12=C3+C4
C
C
C
       DES1=SQRT(C6)
       DET1=2.*C5+1.
C
       E1=C9*C11/DES1
       E2=-C9*C12/DES1
       E3=-E1
       E4=-E2
       E5=1./AR/DET1
       E6=(-C10*C11*C11/2.+C9*C7)/DES1-E1*E1
       E7=-E6
       E8=C10*C11*C12/2./DES1-E1*E2
       E9=-E8
       E10=E8
       E11=(-C10*C12*C12/2.+C9*C7)/DES1-E2*E2
       E12=-E5*E5
C
C**********************************************************************
C      DES=MERIDIAN STRAIN INCREMENT
C      DET=CIRCUMFERENTIAL STRAIN INCREMENT
C      COMPUTATION OF EFFECTIVE STRAIN INCREMENT
C      E1=DERIVATIVE OF MERIDIAN STRAIN INCREMENT WITH RESPECT TO UU(1)
C        =D(DES)/D(UU(1))
C      E2=D(DES)/D(UU(2))
C      E3=D(DES)/D(UU(4))
C      E4=D(DES)/D(UU(5))
```

```
C      E5=D(DET)/D(UU(1))
C      E6=D(E1)/D(UU(1))
C      E7=D(E1)/D(UU(4))
C      E8=D(E1)/D(UU(2))
C      E9=D(E3)/D(UU(2))
C      E10=D(E4)/D(UU(5))
C      E11=D(E5)/D(UU(2))
C**********************************************************************
C
C
       DES=ALOG(DES1)
       DET=ALOG(DET1)
       RVP1=RVALUE+1.
       RVP2=SQRT(2.*RVALUE+1.)
       RVP3=RVP1/RVP2
       RVP4=2.*RVALUE/RVP1
C
C*******************************************************************
C      EFFECTIVE  STRAIN
C      F1=DERIVATIVE OF EFFECTIVE STRAIN INCREMENT WITH RESPECT TO UU(1)
C      F2=WITH RESPECT TO UU(2)
C      F3=WITH RESPECT TO UU(4)
C      F4=WITH RESPECT TO UU(5)
C      F11=D(F1)/D(UU(1))
C      F12=D(F1)/D(UU(2))
C      F13=D(F1)/D(UU(4))
C      F14=D(F1)/D(UU(5))
C      F22=D(F2)/D(UU(2))
C      F23=D(F2)/D(UU(4))
C      F24=D(F2)/D(UU(5))
C      F33=D(F3)/D(UU(4))
C      F34=D(F3)/D(UU(5))
C      F44=D(F4)/D(UU(5))
C*******************************************************************
C
       EFS=DES*DES+DET*DET+RVP4*DES*DET
       EFS1=RVP3*SQRT(EFS)
       EFS2=RVP3/SQRT(EFS)/2.
       EFS3=-RVP3/EFS/SQRT(EFS)/4.
C
C
       D1=(2.*DES+RVP4*DET)*E1+(2.*DET+RVP4*DES)*E5
       D2=(2.*DES+RVP4*DET)*E2
       D3=(2.*DES+RVP4*DET)*E3+(2.*DET+RVP4*DES)*E5
       D4=(2.*DES+RVP4*DET)*E4
       F1=EFS2*D1
       F2=EFS2*D2
       F3=EFS2*D3
       F4=EFS2*D4
C
       F11=EFS3*D1*D1+EFS2*((2.*DES+RVP4*DET)*E6+(2.*DET+RVP4*DES)*E12
      1+(2.*E1+RVP4*E5)*E1+(2.*E5+RVP4*E1)*E5)
       F12=EFS3*D1*D2+EFS2*((2.*DES+RVP4*DET)*E8+(2.*E1+RVP4*E5)*E2)
       F13=EFS3*D1*D3+EFS2*((2.*DES+RVP4*DET)*E7+(2.*E3+RVP4*E5)*E1
      1+(2.*E5+RVP4*E3)*E5+(2.*DET+RVP4*DES)*E12)
       F14=EFS3*D1*D4+EFS2*((2.*DES+RVP4*DET)*E9+(2.*E1+RVP4*E5)*E4)
       F22=EFS3*D2*D2+EFS2*((2.*DES+RVP4*DET)*E11+2.*E2*E2)
       F23=EFS3*D2*D3+EFS2*((2.*DES+RVP4*DET)*E9+(2.*E3+RVP4*E5)*E2)
       F24=EFS3*D2*D4+EFS2*(-(2.*DES+RVP4*DET)*E11+2.*E4*E2)
       F33=EFS3*D3*D3+EFS2*((2.*DES+RVP4*DET)*E6+(2.*E3+RVP4*E5)*E3+(
      12.*E5+RVP4*E3)*E5+(2.*DET+RVP4*DES)*E12)
       F34=EFS3*D3*D4+EFS2*((2.*DES+RVP4*DET)*E10+(2.*E3+RVP4*E5)*E4)
       F44=EFS3*D4*D4+EFS2*((2.*DES+RVP4*DET)*E11+2.*E4*E4)
C
C
       P(1,1)=((S1+S2*EFS1)*F11+S2*F1*F1)*RC*THKL
       P(1,2)=((S1+S2*EFS1)*F12+S2*F2*F1)*RC*THKL
       P(1,4)=((S1+S2*EFS1)*F13+S2*F1*F3)*RC*THKL
       P(1,5)=((S1+S2*EFS1)*F14+S2*F1*F4)*RC*THKL
       P(2,2)=((S1+S2*EFS1)*F22+S2*F2*F2)*RC*THKL
       P(2,4)=((S1+S2*EFS1)*F23+S2*F2*F3)*RC*THKL
       P(2,5)=((S1+S2*EFS1)*F24+S2*F2*F4)*RC*THKL
       P(4,4)=((S1+S2*EFS1)*F33+S2*F3*F3)*RC*THKL
       P(4,5)=((S1+S2*EFS1)*F34+S2*F3*F4)*RC*THKL
       P(5,5)=((S1+S2*EFS1)*F44+S2*F4*F4)*RC*THKL
       P(2,1)=P(1,2)
       P(4,1)=P(1,4)
       P(4,2)=P(2,4)
       P(5,1)=P(1,5)
       P(5,2)=P(2,5)
       P(5,4)=P(4,5)
C
C
       H(1)=-(S1+S2*EFS1)*F1*RC*THKL
       H(2)=-(S1+S2*EFS1)*F2*RC*THKL
```

```
      H(4)=-(S1+S2*EFS1)**E3*EC*THKL
      H(5)=-(S1+S2*EFS1)*E4*EC*THKL
C
C
   71 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE CONDEN(A,B,NEQ,MBAND,N,U)
C
C*************************************************************************
C     PERFORM THE MATRIX CONDENSATION WHEN THE VALUE OF A COMPONENT
C     X IN AX=B IS SPECIFIED EQUAL TO ZERO
C*************************************************************************
C
      DIMENSION B(NEQ),A(NEQ,1)
C
      DO 250 M=2,MBAND
      K=N-M+1
      IF(K) 235,235,230
  230 B(K)=B(K)-A(K,M)*U
      A(K,M)=0.0
  235 K=N+M-1
      IF(NEQ-K) 250,240,240
  240 B(K)=B(K)-A(N,M)*U
      A(N,M)=0.0
  250 CONTINUE
      A(N,1)=1.0
      B(N)=U
      RETURN
C
      END
```

```
      SUBROUTINE MODIFY(CODE,A,B,ALPHA,GAMMA,ETA,NUMNP,NEQ,MBAND,FRNFCE)
      DIMENSION CODE(1),A(NEQ,1),B(1),ALPHA(1),GAMMA(1),ETA(1),FRNFCE(1)
      DO 121 I=1, NUMNP
      IL=3*I
      IZ=IL-1
      IR=IZ-1
      C=CODE(I)
      IF (C.EQ. 1.) GO TO 101
      IF (C.EQ. 2.) GO TO 102
      IF (C.EQ. 3.) GO TO 103
C
      CALL CONDEN(A,B,NEQ,MBAND,IL,0.)
      CONSTA=ALPHA(I)
      CONSTE=ETA(I)
      CONSTG=GAMMA(I)
      IF(C .EQ. 4)GO TO 104
      GO TO 121
C
  101 CONTINUE
      CALL CONDEN(A,B,NEQ,MBAND,IR,0.)
      CALL CONDEN(A,B,NEQ,MBAND,IL,C.)
      GO TO 121
C
  102 CONTINUE
      CALL CONDEN(A,B,NEQ,MBAND,IZ,0.)
      CALL CONDEN(A,B,NEQ,MBAND,IL,C.)
      B(IR)=B(IR)+FRNFCE(I)
      GO TO 121
C
  103 CONTINUE
      CALL CONDEN(A,B,NEQ,MBAND,IR,C.)
      CALL CONDEN(A,B,NEQ,MBAND,IZ,0.)
      CALL CONDEN(A,B,NEQ,MBAND,IL,0.)
      GO TO 121
C
  104 CONTINUE
      CALL BCMIX(A,B,NEQ,MBAND,CONSTA,CONSTE,CONSTG,IR,IZ)
C
  121 CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE TRIA(NN,MM,A)
      DIMENSION A(NN,1)
C
C*******************************************************************************
C     TRIANGULIZATION OF GAUSSIAN ELIMINATION FOR THE SOLUTION
C     OF BANDED SYMMETRIC MATRIX
C*******************************************************************************
C
      N=0
  100 N=N+1
      IF(N.EQ.NN)RETURN
      IF(A(N,1).NE.0.) GO TO 150
      GO TO 100
C
  150 I=N
      MB=MINO(MM,NN-N+1)
C
      DO 260 L=2,MB
      I=I+1
      C=A(N,L)/A(N,1)
      IF(C.EQ.0.0)GO TO 260
      J=0
C
      DO 250 K=L,MB
      J=J+1
  250 A(I,J)=A(I,J)-C*A(N,K)
      A(N,L)=C
  260 CONTINUE
C
      GO TO 100
C
C
      END


      SUBROUTINE BACKS(NN,MM,A,B)
C
C*******************************************************************************
C     BACK SUBSTITUTION FOR SOLUTION OF BANDED SYMMETRIC MATRIX
C*******************************************************************************
C
C
      DIMENSION A(1),B(1)
C
      MMM=MM-1
      N=0
  270 N=N+1
      C=B(N)
      IF(A(N).NE.0.0)B(N)=B(N)/A(N)
      IF(N.EQ.NN)GO TO 300
      IL=N+1
      IH=MINO(NN,N+MMM)
      M=N
      DO 285 I=IL,IH
      M=M+NN
  285 B(I)=B(I)-A(M)*C
      GO TO 270
C
C
  300 IL=N
      N=N-1
      IF(N.EQ.0) RETURN
      IH=MINO(NN,N+MMM)
      M=N
C
      DO 400 I=IL,IH
      M=M+NN
  400 B(N)=B(N)-A(M)*B(I)
      GO TO 300
C
C
      END


      SUBROUTINE HARD(EPS,Y)
      COMMON/MATERL/YYVAL,PRESTN,EEXPNT,PRESTS
C
C*******************************************************************************
C     WORKHARDENING CHARACTERISTIC CURVE
```

```
C*******************************************************************************
C
      YVALUE=YYVAL
      EXPNT=EEXPNT
      Y=YVALUE*(PRESTN+EPS)**EXPNT+PRESTS
      RETURN
      END


      SUBROUTINE HARD2(EPS,Y)
      COMMON/MATERL/YYVAL,PRESTN,EEXPNT,PRESTS
C
C*******************************************************************************
C     COMPUTE WORK HARDENING RATE
C*******************************************************************************
C
      EXPNT=EEXPNT
      YVALUE=YYVAL
      Y=EXPNT*YVALUE*(PRESTN+EPS)**(EXPNT-1.)
      RETURN
      END


      SUBROUTINE  BCMIX(A,B,NEQ,MBAND,CONSTA,CONSTE,CONSTG,IR,IZ)
      DIMENSION A(NEQ,1),B(1)
C
C*******************************************************************************
C     THIS SUBROUTINE HANDLES THE MIXED BOUNDARY CONDITION
C*******************************************************************************
C
      IK=IZ-IR+1
      A(IZ,1)=A(IZ,1)+2.*A(IR,IK)/CONSTA+A(IR,1)/CONSTA/CONSTA
      B(IZ)=B(IZ)+B(IR)/CONSTA-CONSTG*(A(IR,1)/CONSTA+A(IR,IK))
      B(IZ)=B(IZ)+CONSTE
      A(IR,1)=1.0
      A(IR,IK)=0.0
      B(IR)=0.0
C
C
      DO 300 M=2,MBAND
      K=IZ-M+1
      IF( K .LT. 1)GO TO 300
      IF(K .EQ. IR)GO TO 300
      IF(K .GT. IR)GO TO 350
      MM=IR-K+1
      A(K,M)=A(K,M)+A(K,MM)/CONSTA
      B(K)=B(K)-A(K,MM)*CONSTG
      A(K,MM)=0.
      GO TO 300
  350 CONTINUE
C
      MM=K-IR+1
      A(K,M)=A(K,M)+A(IR,MM)/CONSTA
      B(K)=B(K)-A(IR,MM)*CONSTG
      A(IR,MM)=0.
  300 CONTINUE
C
      MB1=MBAND-1
      DO 200 M=2,MB1
      MM=M+1
      A(IZ,M)=A(IZ,M)+A(IR,MM)/CONSTA
      IZZ=IR+M
      B(IZZ)=B(IZZ)-A(IR,MM)*CONSTG
  200 A(IR,MM)=0.0
C
C
      RETURN
      END
```

REFERENCES

1. Lee, C. H., and Kobayashi, S., "New Solutions to Rigid-plastic Deformation Problems Using a Matrix Method," Trans. ASME, J. of Engrg. for Ind., Vol. 95, p. 865, 1973.

2. Lee, C. H., and Kobayashi, S., "Deformation Mechanics and Workability in Upsetting Solid Circular Cylinders," Proc. North Amer. Metalworking Res. Conf., Hamilton, Canada, Vol. 1, p. 185, May 1973.

3. Shah, S. N., Lee, C. H., and Kobayashi, S., "Compression of Tall, Circular, Solid Cylinders Between Parallel Flat Dies," Proc. Int. Conf. Prod. Engr., Tokyo, p. 295, 1974.

4. Shah, S. N., and Kobayashi, S., "Rigid-plastic Analysis of Cold Heating by the Matrix Method," Proc. 15th Int. MTDR Conf., p. 603, 1974.

5. Lee, S. H., and Kobayashi, S., "Rigid-plastic Analysis of Bore Expanding and Flange Drawing with Anisotropic Sheet Metals by Matrix Method," Proc. 15th Int. MTDR Conf., p. 561, 1974.

6. Shah, S. N., and Kobayashi, S., "A Theory on Metal Flow in Axisymmetric Piercing and Extrusion," J. Prod. Engrg., Vol. 1, p. 73, 1977.

7. Chen, C. C., Oh, S. I., and Kobayashi, S., "Ductile Fracture in Axi-symmetric Extrusion and Drawing," USAF Technical Report AFML-TR-77-96, June 1977.

8. Hill, R., "A Variational Principle of Maximum Plastic Work in Classical Plasticity," Quart. J. Mech. Appl. Math. 1, p. 18, 1948.

9. Prager, W., and Hodge, P. G. Jr., Theory of Perfectly Plastic Solids, Dover Publications, 1951.

10. Hill, R., "On the Problem of Uniqueness in the Theory of a Rigid-plastic Solid," Part II, J. Mech. Phys. Solids, Vol. 5, p. 1, 1956.

11. Miles, P., "Bifurcation in Rigid-plastic Materials Under Spherically Symmetric Loading Conditions," J. Mech. Phys. Solids, Vol. 17, p. 303, 1969.

12. Chakrabarty, J., "On Uniqueness and Stability in Rigid-plastic Solids," Int. J. Mech. Sci., Vol. 11, p. 723, 1969.

13. Hill, R., Mathematical Theory of Plasticity, Oxford Press, 1950.

14. Hill, R., "Eigenmodel Deformations in Elastic/Plastic Continua," J. Mech. Phys. Solids, Vol. 15, p. 371, 1967.

15. Malvern, L., Introduction to the Mechanics of a Continuous Medium, Prentice-Hall, 1969.

16. Hill, R., "On the Problem of Uniqueness in the Theory of a Rigid-plastic Solid," Part III, J. Mech. Phys. Solids, Vol. 5, p. 133, 1957.

17. Hill, R., "On the Problem of Uniqueness in the Theory of a Rigid-plastic Solid," Part IV, J. Mech. Phys. Solids, Vol. 5, p. 302, 1957.

18. Hill, R., "Stability of Rigid-plastic Solids," J. Mech. Phys. Solids, Vol. 6, p. 1, 1957.

19. Strang, G., and Fox, G. J., An Analysis of the Finite-element Method, Prentice-Hall, 1973.

20. Zienkiewicz, O. C., The Finite-element Method, McGraw-Hill, 1971.

21. Odell, E. I., and Clausen, W. E., "Numerical Solution of a Deep Drawing Problem," ASME Paper 76-WA/prod-3, 1977.

22. Dahlquist, G., Numerical Methods, Prentice-Hall, 1974.

23. Hill, R., "A Theory of the Plastic Bulging of a Metal Diaphragm by Lateral Pressure," Phil. Mag., Vol. 41, p. 1133, 1950.

24. Woo, D. M., "The Analysis of Axisymmetric Forming of Sheet Metal and the Hydrostatic Bulging Process," Int. J. Mech. Sci., Vol. 6, p. 303, 1964.

25. Yamada, Y., and Yokouchi, Y., "Elastic-plastic Analysis of the Hydraulic Bulge Test by the Membrane Theory," Manuf. Res., Vol. 21, p. 26, 1969 (in Japanese).

26. Wang, N. M., and Shammamy, M. R., "On the Plastic Bulging of a Circular Diaphragm by Hydrostatic Pressure," J. Mech. Phys. Solids, Vol. 17, p. 43, 1969.

27. Shammamy, M. M., and Wang, N. M., "Comparison of Experimental and Theoretical Results for the Hydrostatic Bulging of Circular Sheets," SESA Fall Meeting, 1970.

28. Iseki, H., Jimma, T., and Murota, T., "Finite-element Method of Analysis of the Hydrostatic Bulging of a Sheet Metal," Bull. JSME, Vol. 17, 1974.

29. Wang, N. M., "A Variational Method for Problems of Large Plastic Deformation of Metal Sheets," General Motors Report, 1970.

30. Budiansky, A. B., "Nonlinear Shell Theory," J. of Appl. Mech., Vol. 35, p. 393, 1968.

31. Mellor, P. B., "Stretch Forming Under Fluid Pressure," J. Mech. Phys. Solids, Vol. 5, p. 41, 1956.

32. Loxley, E. M., and Freeman, P., "Some Lubrication Effects in Deep Drawing Operations," J. of Inst. Petr., Vol. 40, p. 299, 1954.

33. Keeler, S. P., and Backofen, W. A., "Plastic Instability and Fracture in Sheets Stretched over Rigid Punches," Trans. ASME, Vol. 56, p. 25, 1963.

34. Swift, H. W., "Plastic Instability Under Plane Stress," J. Mech. Phys. Solids, Vol. 1, p. 1, 1952.

35. Hill, R., "On Discontinuous Plastic States with Special Reference to Localized Necking in Thin Sheets," J. Mech. Phys. Solids, Vol. 1, p. 19, 1952.

36. Marciniak, Z., and Kuczynski, K., "Limit Strains in the Processes of Stretch Forming Sheet Metal," Int. J. Mech. Sci., Vol. 9, p. 609, 1967.

37. Marciniak, Z., Kuczynski, K., and Pokara, T., "Influence of the Plastic Properties of a Material on the Forming Limit Diagram for Sheet Metal in Tension," Int. J. Mech. Sci., Vol. 15, p. 789, 1973.

38. Gosh, A. K., and Hecker, S. S., "Failure in Thin Sheets Stretched over Rigid Punches," General Motors Report, 1974.

39. Kaftanoglu, B., and Alexander, J. M., "On Quasistatic Axisymmetrical Stretch Forming," Int. J. Mech. Sci., Vol. 12, p. 1065, 1970.

40. Chakrabarty, J., "A Theory of Stretch Forming over Hemispherical Punch Heads," Int. J. Mech. Sci., Vol. 12, p. 315, 1970.

41. Woo, D. M., "The Stretch-forming Test," The Engineer, Vol. 220, p. 876, 1965.

42. Wang, N. M., and Gordon, W. J., "On the Stretching of a Circular Sheet by a Hemispherical Punch," General Motors Report, 1968.

43. Wang, N. M., "Large Plastic Deformation of a Circular Sheet Caused by Punch Stretching," General Motors Report, 1969, and J. Appl. Mech., p. 431, 1970.

44. Wifi, A. S., "An Incremental Complete Solution of the Stretch-forming and Deep Drawing of a Circular Blank Using a Hemispherical Punch," Int. J. Mech. Sci., Vol. 18, p. 23, 1976.

45. Lee, C. H., Masaki, S., and Kobayashi, S., "Analysis of Ball Indentation," Int. J. Mech. Sci., Vol. 14, p. 417, 1972.

46. Jarvinen, P. A., "Representation of High Temperature Plastic Behavior of Austenitic and Ferritic Stainless Steel by Empirical Equations," Scand. J. Metallurgy, Vol. 6, 1977.

47. Voce, E., J. of Inst. of Metals, Vol. 74, pp. 537-562, 1948.

48. Gegel, H. L., Private communication, January 1978.

49. Chakrabarty, J., and Mellor, P. B., "A New Approach for Predicting the Limiting Drawing Ratio," IDDRG 5th Biennial Congress, September 30, 1968, la Metallurgia Italiana, p. 791, 1968.

50. El-Sabaie, M. G., and Mellor, P. B., "Plastic Instability Conditions in the Deep Drawing of a Circular Blank of Sheet Metal," Int. J. Mech. Sci., Vol. 4, p. 535, 1972.

51. Budiansky, B., and Wang, N. M., "On the Swift Cup Test," J. Mech. Phys. Solids, Vol. 14, p. 357, 1966.

52. Chung, S. Y., and Swift, H. W., "Cup Drawing from a Flat Blank," Proc. Instr. Mech. Engrs., Vol. 165, p. 199, 1951.

53. Woo, D. M., "Analysis of the Cup Drawing Processes," J. Mech. Engrs., Vol. 6, p. 116, 1964.

54. Lee, P. K., Choi, C. Y., and Hsu, T. C., "Effect of Drawing on Formability in Axisymmetrical Sheet Metal Forming," J. of Eng. Ind., p. 925, 1973.

55. Hsu, T. C., Dowle, W. R., Choi, C. Y., and Lee, P. K., "Strain Histories and Strain Distributions in a Cup Drawing Operation," J. of Eng. Ind., p. 461, 1971.

56. El-Sebaie, M. G., and Mellor, P. B., "Double Operation Deep Drawing," Int. J. Mech. Sci., Vol. 15, p. 945, 1973.

57. Chiang, D. C., and Kobayashi, Shiro, "The Effect of Anisotropy and Work-hardening Characteristics on the Stress and Strain Distribution in Deep Drawing," ASME paper 66-prod-3, 1966.

58. Woo, D. M., "On the Complete Solution of the Deep Drawing Problem," Int. J. Mech. Sci., Vol. 10, p. 83, 1968.

59. Levy, S., Shih, C. F., Wilkinson, J. P. D., Stine, P., and McWilson, R. C., "Analysis of Sheet Metal Forming to Axisymmetric Shapes," General Electric Report No. 77CRD257, December 1977, Schenectady, New York.

☆U.S.Government Printing Office: 1979 — 657-002/490